BAB II

TINJAUAN PUSTAKA

2.1 PENELITIAN TERDAHULU

Sebagai bahan acuan untuk penyusunan skripsi ini penulis memaparkan hasil dari penelitian terdahulu yang pernah dilakukan, diantaranya :

Penelitian karya tulis ilmiah yang dilakukan oleh Octovensa Purba dari Universitas Advent Indonesia dengan judul Analisa Alat Pertahanan Dengan Menggunakan Honeyd Terhadap Serangan Buffer Overflow Pada Linux Ubuntu 12.04, menjalaskan bahwa Honeyd adalah salah satu dari banyak aplikasi Honeypot yang ada saat ini, dapat menemulasikan virtual host dan servis mirip dengan yang aslinya. Honeyd dapat mendeteksi semua serangan yang ditujukan kepada virtual host yang diemulasikan oleh Honeyd.

Dalam penelitian itu disebutkan bahwa berdasarkan hasil penelitian penulis, Honeyd tidak dapat mendeteksi serangan Buffer Overflow yang ditujukan kepada virtual host yang sudah diemulasikan oleh karena Honeyd adalah Low Interaction Honeypot sehingga interaksi yang dilakukan tidak dapat dideteksi oleh Honeyd. Oleh karena penulis menggunakan aplikasi tambahan wireshark sehingga dapat mendeteksi serangan yang diarahkan kepada virtual host. Honeyd adalah salah satu Low Interaction Honeypot sehingga memiliki keterbatasan di dalam mensimulasikan interaksi dari virtual host. (Octovensa purba, Universitas Advent Indonesia)

2.2 DASAR TEORI

Pada dasar teori ini akan dibahas mengenai jaringan komputer, Jenis-jenis Jaringan Komputer, Referensi Model OSI LAYER, Network Security, IDS (Intrusion Detection System), Honeypot, Honeyd, DDoS (Distributed Denial of Service) , HTTP Flood dan TCP Flood.

2.2.1 Jaringan Komputer

Jaringan komputer adalah sekumpulan komputer, serta perangkat-perangkat lain pendukung komputer yang saling terhubung dalam suatu kesatuan. Media jaringan komputer dapat melalui kabel-kabel atau tanpa kabel sehingga memungkinkan pengguna jaringan komputer dapat saling melakukan pertukaran informasi, seperti dokumen dan data, dapat juga melakukan pencetakan pada printer yang sama dan bersama-sama memakai perangkat keras dan perangkat lunak yang terhubung dengan jaringan. Setiap komputer, ataupun perangkat-perangkat yang terhubung dalam suatu jaringan disebut dengan node. Dalam sebuah jaringan komputer dapat mempunyai dua, puluhan, ribuan atau bahkan jutaan node. Jaringan Komputer juga memiliki arti sebagai sekelompok komputer otonom yang saling berhubungan antara satu dengan lainnya menggunakan protokol komunikasi melalui media komunikasi sehingga dapat saling berbagi informasi, program – program, penggunaan bersama perangkat keras seperti Printer, Harddisk, dan sebagainya. Selain itu jaringan komputer bisa diartikan sebagai kumpulan sejumlah terminal komunikasi yang berada diberbagai lokasi yang terdiri dari lebih satu komputer yang saling berhubungan. (Syafrizal. Melwin, 2005)

Manfaat yang didapat dalam membangun jaringan komputer, yaitu :

1. Sharing Resources

Sharing resources bertujuan agar seluruh program, peralatan lainnya dapat dimanfaatkan oleh setiap orang yang ada pada jaringan komputer tanpa terpengaruh oleh lokasi maupun pengaruh dari pemakai.

2. Media Komunikasi

Jaringan komputer memungkinkan terjadinya komunikasi antar pengguna, baik untuk teleconference maupun untuk mengirim pesan atau informasi yang penting lainnya.

3. Integrasi Data

Jaringan komputer dapat mencegah ketergantungan pada komputer pusat, karena setiap proses data tidak harus dilakukan pada satu komputer saja, melainkan dapat didistribusikan ke tempat lainnya. Oleh sebab inilah maka dapat terbentuk data yang terintegrasi, memudahkan pemakai untuk memperoleh dan mengolah informasi setiap saat.

4. Pengembangan dan Pemeliharaan

Pengembangan peralatan dapat dilakukan dengan mudah dan menghemat biaya, karena setiap pembelian komponenennya seperti Printer, maka tidak perlu membeli Printer sejumlah komputer yang ada tetapi cukup satu buah karena Printer itu dapat digunakan secara bersama – sama. Jaringan komputer juga memudahkan pemakai dalam merawat Harddisk dan peralatan lainnya, misalnya

untuk memberikan perlindungan terhadap serangan virus maka pemakai cukup memusatkan perhatian pada Harddisk yang ada pada komputer pusat.

5. Keamanan Data

Sistem Jaringan Komputer dapat memberikan perlindungan terhadap data. Karena pemberian dan pengaturan hak akses kepada para pemakai, serta teknik perlindungan terhadap harddisk sehingga data mendapatkan perlindungan yang efektif.

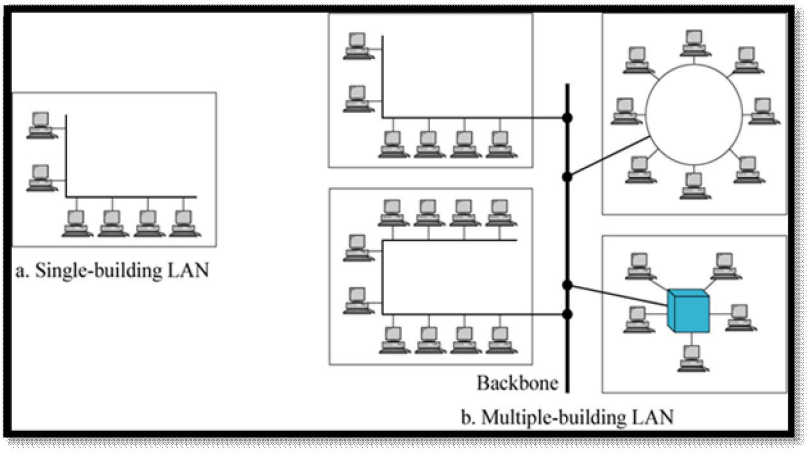
6. Sumber Daya Lebih Efisien dan Informasi Terkini

Dengan pemakaian sumber daya secara bersama – sama, akan mendapatkan hasil yang maksimal dan kualitas yang tinggi. Selain itu data atau informasi yang diakses selalu terbaru, karena setiap ada perubahan yang terjadi dapat segera langsung diketahui oleh setiap pemakai.

2.2.2 Jenis-Jenis Jaringan Komputer

2.2.2.1 LAN (Local Area Network)

Sebuah LAN, adalah jaringan yang dibatasi oleh area yang relatif kecil, umumnya dibatasi oleh area lingkungan, seperti sebuah kantor pada sebuah gedung, atau tiap-tiap ruangan pada sebuah sekolah. Biasanya jarak antar node tidak lebih jauh dari sekitar 200 meter. Berikut pada gambar 2.1 merupakan contoh gambar dari Local Area Network.

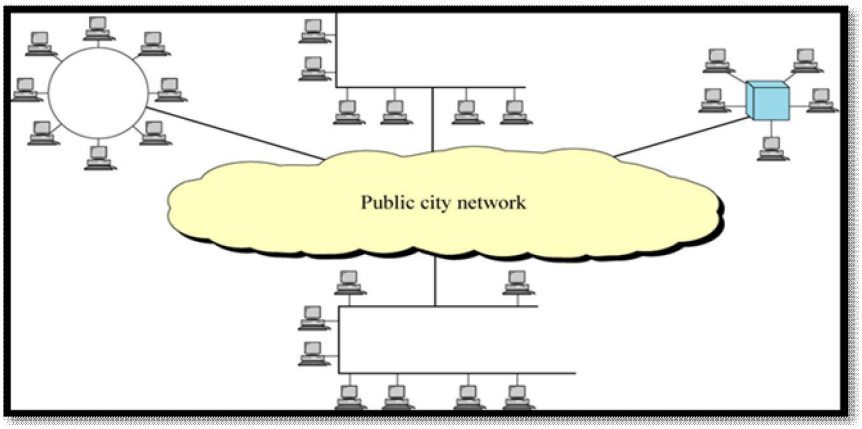


Gambar 2.1 LAN (Local Area Network)

Sumber : (Syafrizal. Melwin, 2005)

2.2.2.2 MAN (Metropolitan Area Network)

Sebuah MAN, biasanya meliputi area yang lebih besar dari LAN, misalnya antar gedung dalam suatu daerah (wilayah seperti propinsi atau negara bagian). Dalam hal ini jaringan menghubungkan beberapa buah jaringan kecil ke dalam lingkungan area yang lebih besar, sebagai contoh yaitu: jaringan beberapa kantor cabang sebuah bank didalam sebuah kota besar yang dihubungkan antara satu dengan lainnya.

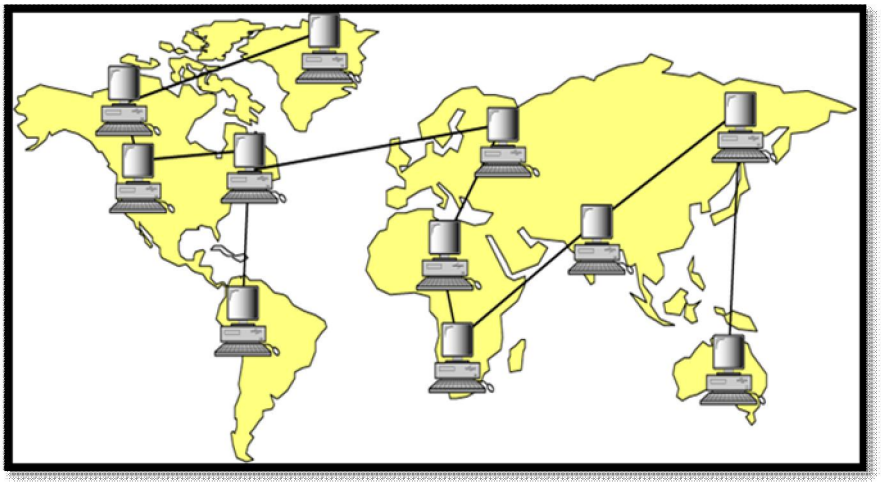


Gambar 2.2 MAN (Metropolitan Area Network)

Sumber : (Syafrizal. Melwin, 2005)

2.2.2.3 WAN (Wide Area Network)

Wide Area Network (WAN) adalah jaringan yang biasanya sudah menggunakan media wireless, sarana satelit ataupun kabel serat optic, karena jangkauannya yang lebih luas, bukan hanya meliputi satu kota atau antar kota dalam suatu wilayah, tetapi mulai menjangkau area/wilayah otoritas negara lain. Sebagai contoh jaringan komputer kantor City Bank yang ada di Indonesia ataupun yang ada di negara lain, yang saling berhubungan, jaringan ATM Master Card, Visa Card atau Cirrus yang tersebar diseluruh dunia dan lain-lain. Biasanya WAN lebih rumit dan sangat kompleks bila dibandingkan LAN maupun MAN. Menggunakan banyak sarana untuk menghubungkan antara LAN dan WAN kedalam komunikasi global seperti internet, meski demikian antara LAN, MAN dan WAN tidak banyak berbeda dalam beberapa hal, hanya lingkup areanya saja yang berbeda satu dengan yang lain.



Gambar 2.3 WAN (Wide Area Network)

Sumber : (Syafrizal, Melwin. 2005)

2.2.3 Referensi Model OSI Layer

Model ini disebut OSI (Open System Interconnection) Reference Model, karena model ini ditujukan untuk pengkoneksian open system. Dikembangkan oleh International Organization for Standardization (ISO) pada tahun 1984. Open System dapat diartikan sebagai suatu sistem yang terbuka untuk berkomunikasi dengan sistem-sistem lainnya. Untuk ringkasnya, kita akan menyebut model tersebut sebagai model OSI saja. OSI menggambarkan bagaimana informasi dari suatu software aplikasi pada sebuah komputer berpindah melewati sebuah media jaringan ke suatu software aplikasi di komputer lain.

Model OSI menyediakan secara konseptual kerangka kerja untuk komunikasi antar komputer, tetapi model ini bukan merupakan metoda komunikasi. Sebenarnya komunikasi dapat terjadi karena menggunakan protokol komunikasi. Dalam konteks jaringan (komunikasi data), sebuah protokol adalah suatu aturan formal dan kesepakatan yang menentukan bagaimana komputer bertukar informasi melewati sebuah media jaringan. Sebuah protokol Mengimplementasikan salah satu atau lebih lapisan-lapisan OSI.

Model OSI secara konseptual terbagi ke dalam 7 lapisan dimana masing-masing lapisan memiliki fungsi jaringan yang spesifik, seperti yang dijelaskan oleh gambar diatas (tanpa media fisik). Model ini diciptakan berdasarkan sebuah proposal yang dibuat oleh the International Standards Organization (ISO) sebagai langkah awal menuju standarisasi protokol internasional yang digunakan pada berbagai layer . Prinsip-prinsip yang digunakan bagi ketujuh layer tersebut adalah :

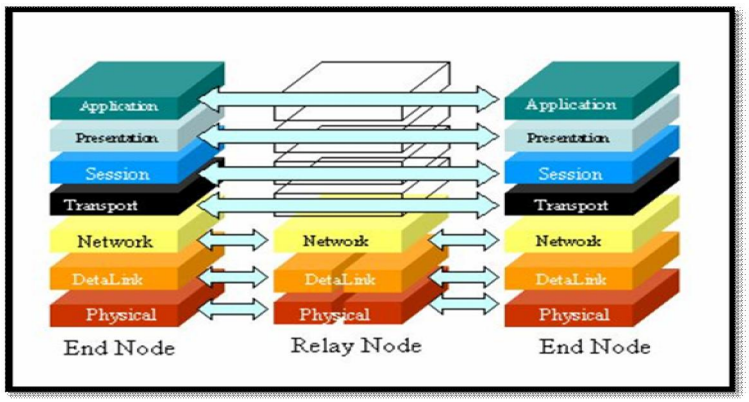
i. Sebuah layer harus dibuat bila diperlukan tingkat abstraksi yang berbeda.

ii. Setiap layer harus memiliki fungsi-fungsi tertentu.

iii. Fungsi setiap layer harus dipilih dengan teliti sesuai dengan ketentuan standar protokol internasional.

iv. Batas-batas layer diusahakan agar meminimalkan aliran informasi yang melewati interface.

v. Jumlah layer harus cukup banyak, sehingga fungsi-fungsi yang berbeda tidak perlu disatukan dalam satu layer diluar keperluannya. Akan tetapi jumlah layer juga harus diusahakan sesedikit mungkin sehingga arsitektur jaringan tidak menjadi sulit dipakai.



Gambar 2.4 Model Referensi OSI LAYER

Sumber : (Syafrizal. Melwin, 2005)

2.2.4 Network Security (Keamanan Jaringan)

Keamanan jaringan (Network Security) dalam jaringan komputer sangat penting dilakukan untuk memonitor akses jaringan dan mencegah penyalahgunaan sumber daya jaringan yang tidak sah. Tugas keamanan jaringan dikontrol oleh administrator jaringan. Segi-segi keamanan didefinisikan dari kelima point.

a. Confidentiality (kerahasiaan) artinya Mensyaratkan bahwa informasi (data) hanya bisa diakses oleh pihak yang memiliki wewenang.
b. Integrity (integritas) artinya Mensyaratkan bahwa informasi hanya dapat diubah oleh pihak yang memiliki wewenang.
c. Availability (tersedianya) artinya Mensyaratkan bahwa informasi tersedia untuk pihak yang memiliki wewenang ketika dibutuhkan.
d. Authentication (otentikasi) artinya Mensyaratkan bahwa pengirim suatu informasi dapat diidentifikasi dengan benar dan ada jaminan bahwa identitas yang didapat tidak palsu.
e. Non-repudiation Mensyaratkan bahwa baik pengirim maupun penerima informasi tidak dapat menyangkal pengiriman dan penerimaan. Sebagai contoh, seseorang yang mengirimkan email untuk memesan barang tidak dapat menyangkal bahwa dia telah mengirimkan email tersebut. (Referens 1, Anonim, 2013).

2.2.5 IDS (Intrusion Detection System)

Intrusion Detection System (IDS) atau sistem pendeteksian penyusupan merupakan sebuah konsep canggih yang melibatkan beberapa teknologi yang berbeda. Boleh dikatakan bahwa IDS sudah menjadi sepenting firewall untuk security network, dan banyak perbedaan antara IDS dan firewall yang telah menjadi kabur. Dulunya, IDS mencakup hal tentang menganalisis lalu lintas network untuk mencari bukti dari sebuah serangan. Namun saat ini IDS semakin diperluas sehingga juga menyangkut tentang scanning log-log akses dan menganalisis karakteristik-karakteristik dari file-file untuk mengetahui apakah file-file tersebut telah diserang. IDS juga telah diperluas ke konsep Honeypot

yaitu sebuah network palsu yang digunakan untuk menarik dan mengalihkan perhatian para cracker dari network yang sesungguhnya, dan pada saat yang sama melakukan pemantauan terhadap aksi-aksi mereka di network palsu itu. Tipe-tipe dari IDS akan dijelaskan pada sub bab 2.2.5.1 sampai 2.2.5.5.

2.2.5.1 Network Intrusion Detection System (NIDS)

Sistem pendeteksian penyusupan network. Menganalisis paket di sebuah network dan mencoba untuk menentukan apakah seorang cracker sedang mencoba untuk masuk ke dalam sebuah sistem atau menyebabkan sebuah serangan denial of service (DoS). Sebuah NIDS bisanya berjalan pada sebuah hub atau sebuah router, dan menganalisis semua lalu lintas network yang mengalir melalui alat tersebut.

2.2.5.2 Host Intrusion Detection System (HIDS)

Sistem pendeteksian penyusupan host. Sama seperti NIDS, sebuah HIDS menganalisis lalu lintas network yang dikirimkan menuju dan dari sebuah mesin tunggal. Sebagian besar dari NIDS komersial saat ini biasanya memiliki suatu insur HIDS, dan sistem-sistem ini disebut hybrid ID.

2.2.5.3 System Integrity Verifier (SIV)

Alat untuk verifikasi integritas sistem. Melacak file-file sistem yang kritikal dan memberitahukan kepada administrator pada saat file-file tersebut diubah (biasanya oleh seorang cracker yang mencoba untuk mengganti file yang valid dengan sebuah Trojan horse).

2.2.5.4 Honeypot

Sebuah sistem pura-pura yang memiliki servis-servis yang tidak nyata, dengan vulnerability-vulnerability yang sudah diketahui untuk menarik perhatian para hacker atau mengalihkan perhatian mereka dari sistem yang sebenarnya.

2.2.5.5 Log File Monitor (LFM)

Pemantau file log. Membaca log-log yang dihasilkan oleh servis-servis network yang mencari pola-pola serangan. (Brenton, Chris., Hunt, Cameron. 2005)

2.2.6 Honeypot

Dalam bahasa sederhana, Honeypot adalah sistem atau komputer yang sengaja dikorbankan untuk menjadi target serangan hacker. Oleh sebab itu setiap interaksi dengan Honeypot patut diduga sebagai aktivitas penyusupan. Misal, jika ada orang yang melakukan scanning jaringan untuk mencari komputer yang vulnerable (rentan), saat ia mencoba koneksi ke Honeypot tersebut, maka Honeypot akan mendeteksi dan mencatatnya, karena seharusnya tidak ada User yang berintereaksi dengan Honeypot. Keunggulan hacker adalah anonimitas. Honeypot merupakan senjata orang-orang baik yang membuat situasi menjadi lebih imbang. Tidak seperti IDS (Intursion Detection System) atau firewall, Honeypot tidak menyelesaikan suatu masalah tertentu, tetapi memiliki kontribusi terhadap keseluruhan keamanan. Nilai kontribusi bergantung pada kita mempergunakannya. Jadi meski tidak secara langsung mencegah serangan seperti firewall, tetapi bisa mengurangi intensitas serangan pada server sungguhan. Honeypot memang hanya mempunyai manfaat kecil pada pencegahan, tetapi

sangat berguna untuk mendeteksi serangan, perlu diingat bahwa firewall juga tidak bisa menghilangkan serangan, tetapi hanya memperkecil resiko serangan.

Honeypot mengumpulkan sedikit data tetapi dengan nilai yang tinggi sehingga memungkinkan analisis dan respon yang cepat. Contohnya, Honeypot project suatu grup penelitian Honeypot mengumpulkan kurang dari 1 MB data per hari. Volume data tidak sebanyak log pada sistem firewall atau IDS. Memang konfigurasinya bisa memakan waktu, tetapi merupakan cara yang menarik bagi kita untuk mempelajarinya. Bayangkan bermacam potensi lain pemanfaatan honeypot, seperti memonitor pembajakan software dan tool hacking yang paling populer. Kesederhanaan pengunaan Honeypot memudahkan konfigurasi pemanfaatanya, meski juga ada yang kompleks untuk keperluan penelitian. Makin sederhana Honeypot, semakin kecil resikonya. (Utdirartatmo, Firrar. 2005).

2.2.6.1 Bentuk Honeypot

Honeypot memiliki bermacam bentuk dan ukuran, dari yang sederhana semacam emulasi sejumlah service, sampai suatu jaringan yang didesain untuk di hack. Dari hanya sekedar alarm pendeteksi penyusup sampai untuk penelitian motivasi hacking. Honeypot bisa berjalan pada bermacam sistem operasi dan menjalankan bermacam service. Konfigurasi service menunjukan ketersediannya pada suatu usaha probing atau compromise pada sistem, dalam hal ini Honeypot dibedakan menjadi:

a. High Interaction: Mensimulasikan semua aspek dari suatu sistem operasi. Bisa berisiko mengalami compromise yang memungkinkan

untuk dipakai penyusup untuk memperoleh akses penuh ke jaringan, atau melakukan serangan selanjutnya.

b. Low Interaction: Mensimulasikan hanya sejumlah bagian layanan, seperti network stack. Penyusup tidak bisa memperoleh akses penuh ke Honeypot. Meski terbatas, tetapi berguna untuk memperoleh informasi mengenai probing jaringan atau aktivitas worm. Mereka juga bisa dipergunakan untuk menganalisis spammer atau melakukan countermeasure pada worm.

Secara singkat dapat dinyatakan dalam tabel berikut:.

Low-interaction Mengemulasi sistem operasi dan service	High-interaction Sistem operasi dan service sunguhan tanpa emulasi
• Mudah di-install dan deploy. Konfigurasi software biasanya sederhana • Risiko minimal, emulasi mengontrol apa yang bisa dilakukan penyusup • Menangkap jumlah informasi terbatas	• Menangkap informasi lebih banyak • Bisa cukup kompleks • Resiko tinggi, penyusup bisa berinteraksi dengan sistem operasi sungguhan

Tabel 2.1 Dua bentuk Honeypot

Selain itu Honeypot juga bisa dibedakan ke dalam:

a. Physical: Mesin sungguhan dalam jaringan dengan alamat IP sendiri.

b. Virtual: Disimulasikan oleh mesin lain yang berespon pada traffic jaringan yang dikirim ke virtual Honeypot.

Umumnya physical Honeypot adalah high-interaction, maka bisa saja mengalami compromise sepenuhnya. Selain itu, lebih sulit dikelola. Salah satu contohnya adalah Honeynet. Untuk alamat IP yang banyak, lebih praktis memakai virtual Honeypot.

Saat mengumpulkan informasi mengenai suatu usaha penyusupan, jumlah Honeypot yang ada mempengaruhi tingkat akurasi data. Misal, untuk mengukur aktivitas worm berbasis HTTP, kita bisa mengenalinya hanya setelah mereka melakukan suatu TCP handshake yang lengkap. Tetapi kebanyakan usaha koneksi akan tidak terjawab karena worm melakukan kontak pada alamat IP yang dipilih secara acak. Honeypot bisa menangkap beban worm dengan konfigurasi sebagai web server. Lebih banyak Honeypot yang ada, berarti lebih banyak kemungkinan akan dihubungi oleh worm. (Utdirartatmo, Firrar. 2005).

2.2.6.2 Deteksi Efektif Dengan Honeypot

Honeypot membuat tugas deteksi menjadi lebih sederhana, efektif, dan murah. Konsep Honeypot sangat mudah dipahami dan diimplementasikan. Honeypot tidak memiliki rule untuk update atau perubahan, dan tidak ada algoritma lanjut yang diperlukan untuk menganalisa traffic jaringan. Honeypot juga merupakan solusi yang sanagat efektif dan murah. Dengan kecilnya keperluan sumber daya, pemeliharaan, dan analisa, maka bisa mengurangi biaya. User tidak perlu membeli komputer khusus berkinerja tinggi yang mahal, cukup dengan pentium lama dan card LAN 10/100 sudah mencukupi untuk jaringan kelas C. Penghematan lebih banyak berasal dari sumber daya manusia. Honeypot mengurangi secara dramatis jumlah informasi yang perlu dikumpulkan,

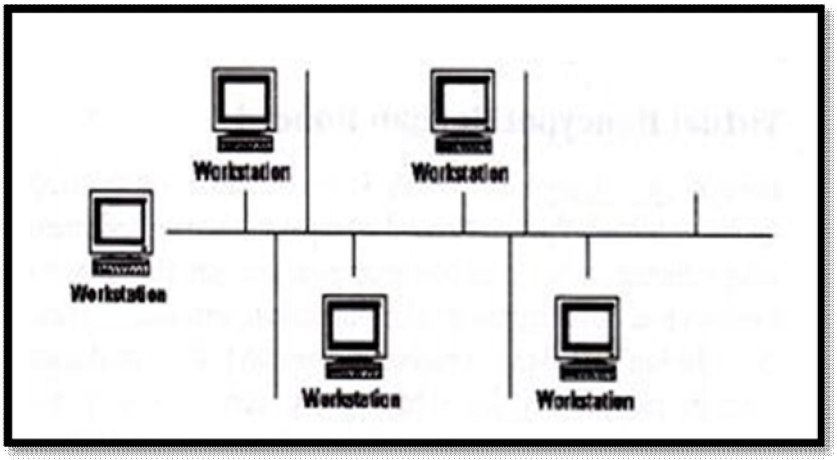
dihubungkan, dan disimpan. Ini menghemat jam kerja, sehingga tenaga keamanan bisa berfokus pada aktivitas penting lainnya, semacam patching. Honeypot juga mengatasi sejumlah kegagalan NIDS, semacam deteksi serangan baru, atau lingkungan terenkripsi atau protokol IPv6.

Tentu saja seperti halnya teknologi manapun, honeypopt juga memilii kekurangan. Kekurangan terbesar berasal dari keterbatasan pandangan, karena hanya bisa menangkap aktivitas yang diarahkan pada mereka, dan tidak akan mengangkap serangan pada sistem yang lain. Misal, jika web server diserang, Honeypot tidak akan mengetahuinya (kecuali penyerang memakai web server untuk melakukan scan pada Honeypot). Disarankan untuk tidak memakai Honeypot sebagai pengganti teknologi deteksi yang ada, tetapi untuk saling bekerjasama dan melengkapi strategi keamanan jaringan. (Utdirartatmo, Firrar. 2005).

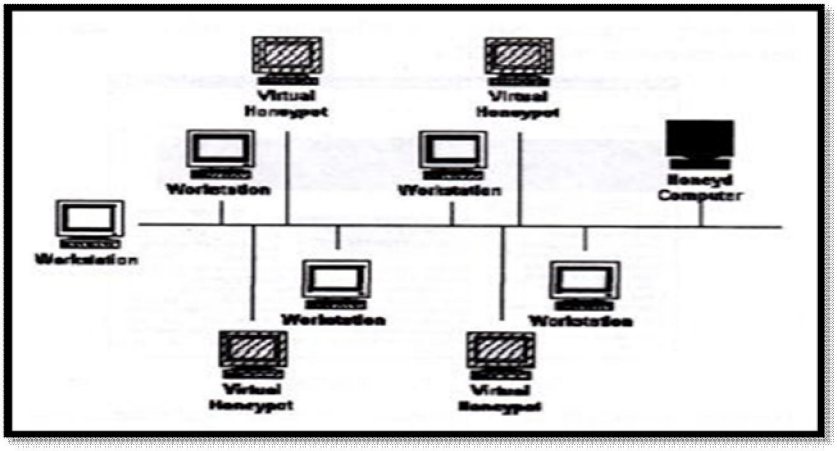
2.2.7 Honeyd

Honeyd adalah Honeypot open source yang di tulis oleh Neils Provos. Honeyd merupakan daemon sederhana yang membuat suatu virtual host tetap pada jaringan. Host tersebut nantinya bisa dikonfigurasi untuk menjalankan bermacam service. Kepribadian TCP-nya disebut personality bisa diadaptasi sehingga nampak berjalan sebagai suatu versi sistem operasi tertentu, untuk mengelabui scanner fingerprint semacam xprobe atau nmap. Sebenarnya Honeyd cukup powerfull dan menyediakan fitur yang lengkap. Sayang konfigurasinya tidak mudah, dan belum memiliki interface GUI. Honeyd memungkinkan suatu host mengklaim sejumlah alamat IP untuk disimulasikan sehingga nampak seolah

suatu jaringan. Sederhananya, misal mempunyai LAN dengan IP 192.168.0.1 – 25, tidak semua alamat IP tersebut dipakai oleh komputer aktif. Disini alamat-alamat IP yang tidak terpakai bisa dimonitor oleh Honeypot. Karena usaha akses, probing dan scanning ke unused IP tersebut menandakan usaha akses ilegal.



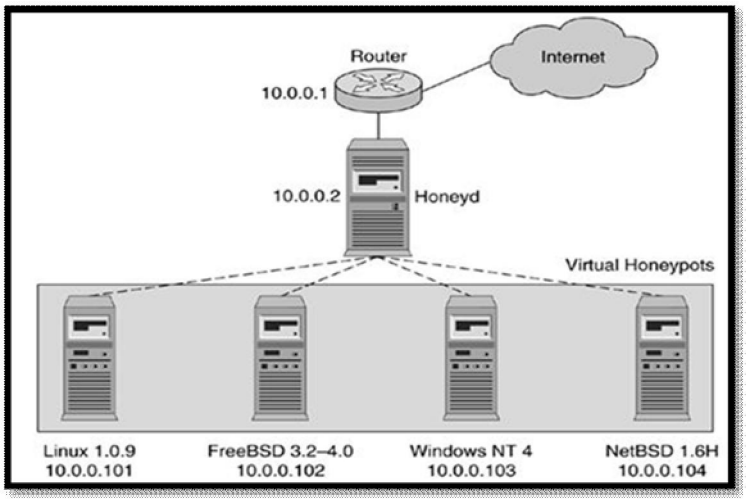
Gambar 2.5 Jaringan Dengan Sejumlah Unused IP



Gambar 2.6 Honeyd bisa memonitor unused IP.

Virtual host yang ada bisa di-ping atau trace route. Tipe service pada virtual host bisa disimulasikan dengan memakai file konfigurasi sederhana. Honyed memiliki personality yang berperilaku soalah-olah paket yang dikirimnya berasal dari suatu sistem operasi. Daripada mensimulasikan semua aspek dari sistem operasi,

Honeyd memilih berada pada tingkatan network stack. Honeyd mensimulasikan service TCP dan UDP, serta berespon dengan tepat pada paket ICMP.



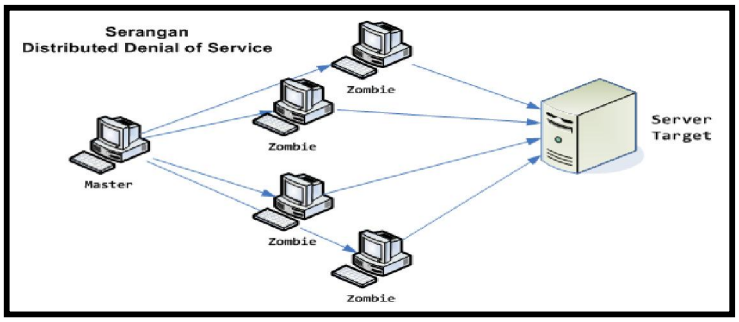
Gambar 2.7 Contoh virtual Honeypot dengan bermacam sistem operasi.

Sejumlah fitur yang ada dari Honeyd:

1. Simulasi ratusan virtual host pada saat bersamaan (tergantung pada kemampuan komputer yang dimiliki).
2. Konfigurasi bermacam service termasuk proxy connect, passive fingerprinting, dan random sampling.
3. Simulasi sistem operasi pada tingkat TCP/IP stack: mengelabuhi Nmap dan Xprobe, fragment reassembly, FIN-scan.
4. Simulasi bermacam topologi routing: Latency & packet loss, Assymetric routing, integrasi mesin fisik, Distributed Honeyd via GRE tunneling.
5. Subsystem virtualization: Menjalankan aplikasi UNIX sesungguhnya di bawah alamat IP virtual Honeyd (web server, ftp server, dst), dynamic port binding dalam virtual address space. (Utdirartatmo, Firrar. 2005)

2.2.8 DDoS (Distributed Denial of Service)

Distributed Denial of Service (DDoS) adalah suatu serangan Denial of Service (Dos) yang menggunakan banyak host penyerang, baik itu menggunakan komputer yang didedikasikan untuk melakukan penyerangan atau komputer yang dipaksa menjadi zombie untuk menyerang satu buah host target dalam sebuah jaringan. Serangan Denial of Service klasik bersifat satu lawan satu, sehingga dibutuhkan sebuah host yang kuat baik itu dari kekuatan pemrosesan atau sistem operasinya demi membanjiri lalu lintas klien yang valid untuk mengakses layanan jaringan pada server yang dijadikan target serangan. Serangan DDoS ini menggunakan teknik canggih dibandingkan dengan serangan Denial of Service yang klasik, yakni dengan meningkatkan serangan beberapa kali dengan menggunakan beberapa buah komputer sekaligus, sehingga dapat mengakibatkan server atau keseluruhan segmen jaringan dapat menjadi tidak berguna sama sekali bagi klien. (Referens 2, Anonim, 2013).



Gambar 2.8 Skema Serangan DDoS (Distributed Denial of Service)

Serangan DDoS pertama kali muncul pada tahun 1999, tiga tahun setelah serangan Denial of Service yang klasik muncul, dengan menggunakan serangan SYN Flooding, yang mengakibatkan beberapa server web di internet mengalami downtime. Pada awal februari 2000, sebuah serangan yang besar dilakukan

sehingga beberapa situs web terkenal seperti Amazon, CNN, eBay, dan Yahoo! Mengalami downtime selama beberapa jam. teori dan praktik untuk melakukan serangan DDoS justru sederhana, yakni sebagai berikut :

1. Menjalankan tool biasanya berupa program perangkat lunak kecil yang secara otomatis akan memindai jaringan untuk menemukan host-host yang rentan (vulnerable) yang terkoneksi ke Internet. Setelah host yang rentan ditemukan, tool tersebut dapat menginstalasikan salah satu jenis dari Trojan Horse yang disebut sebagai DDoS Trojan, yang akan mengakibatkan host tersebut menjadi zombie yang dapat dikontrol secara jarak jauh oleh sebuah komputer master yang digunakan oleh penyerang asli untuk melancarkan serangan. Beberapa tool atau software yang digunakan untuk melakukan serangan seperti ini adalah TFN, TFN2K, Trinoo, dan Stacheldraht, yang dapat diunduh secara bebas di Internet.

2. Ketika penyerang merasa telah mendapatkan jumlah host yang cukup sebagai zombie untuk melakukan penyerangan, penyerang akan menggunakan komputer master untuk memberikan sinyal penyerangan terhadap jaringan target atau host target. Serangan ini umumnya dilakukan dengan menggunakan beberapa bentuk SYN Flood atau skema serangan DoS yang sederhana, tapi karena dilakukan oleh banyak host zombie, maka jumlah lalu lintas jaringan yang diciptakan adalah sangat besar, sehingga memakan habis semua sumber daya Transmission Control Protocol yang terdapat di dalam komputer atau jaringan target.

Hampir semua platfrom komputer dapat dibajak sebagai sebuah zombie untuk melakukan serangan seperti ini. Sistem-sistem populer, semacam Solaris, Linux,

Microsoft Windows dan beberapa varian UNIX dapat menjadi zombie, jika memang sistem tersebut atau aplikasi yang berjalan di atasnya memiliki kelemahan yang dieksploitasi oleh penyerang.

Beberapa contoh serangan DoS :

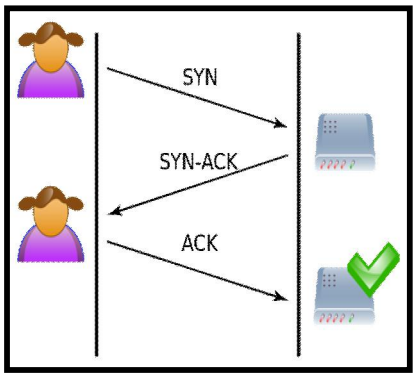
a. Serangan Buffer Overflow, mengirimkan data yang melebihi kapasitas sistem, misalnya paket ICMP yang berukuran sangat besar.
b. Serangan SYN, mengirimkan data TCP SYN dengan alamat palsu.
c. Serangan Teardrop, mengirimkan paket IP dengan nilai offset yang membingungkan
d. Serangan Smurf, mengirimkan paket ICMP bervolume besar dengan alamat host lain. (Referens 2, Anonim, 2013).

2.2.9 HTTP (Hypertext Transfer Protocol) Flood

Lebih dari 80% dari serangan DDoS modern adalah serangan DDoS HTTP Flood, tidak seperti kebanyakan serangan jaringan yang membanjiri sumber daya komputasi dengan paket yang tidak valid, serangan HTTP Flood terlihat seperti permintaan web HTTP nyata. Ribuan atau jutaan menyerang klien membanjiri server web dengan sejumlah besar permintaan. Dua variasi utama dari serangan HTTP Flood berbeda dalam konten yang diminta. Yang paling umum , serangan dasar hanya mengulangi permintaan yang sama berulang-ulang lagi. (David Holmes : 7-8).

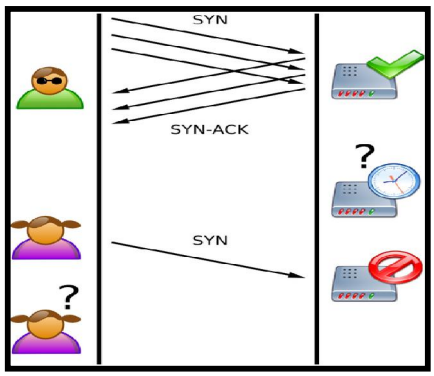
2.2.10 TCP (Transmission Control Protocol) Flood

Paket-paket SYN adalah salah satu jenis paket dalam protokol Transmission Control Protocol (TCP) yang dapat digunakan untuk membuat koneksi antara dua host dan dikirimkan oleh host yang hendak membuat koneksi, sebagai langkah pertama pembuatan koneksi dalam proses TCP Three-way Handshake.



Gambar 2.9 Skema 3-Way-Handshake

Dalam sebuah serangan SYN Flooding, Penyerang akan mengirimkan paket-paket SYN ke dalam port-port yang sedang berada dalam keadaan listinening yang berada dalam host target. Normalnya, paket-paket SYN yang dikirimkan berisi alamat sumber yang menunjukan sistem aktual, tetapi paket-paket SYN dalam serangan ini didesain sedemikian rupa, sehingga paket-paket tersebut memiliki alamat sumber yang tidak menunjukan sistem aktual.



Gambar 2.10 Skema TCP SYN Flood.

Ketika target menerima paket-paket SYN/ACK yang ditujukan kepada alamat tercantum di dalam SYN packet yang ia terima yang berarti sistem tersebut tidak ada secara aktual dan kemudian akan menunggu paket Acknowledgment (ACK) sebagai balasan untuk melengkapi proses pembuatan koneksi. Tetapi, karena alamat sumber dalam paket SYN yang dikirimkan oleh penyerang tidaklah valid, port yang menjadi terget serangan akan menunggu hingga waktu pembuatan koneksi kadaluwarsa atau timed-out.(Referens 3, Anonim, 2013).

BAB III

METODOLOGI PENELITIAN

3.1 ALUR PENELITIAN

Dalam bab ini akan membahas tentang penyelesaian masalah yang sudah dipaparkan dalam rumusan masalah. Untuk rancangan jaringan akan dibuat skema rancangan jaringannya agar memudahkan dalam mengimplementasikannya. Dalam penelitian skripsi ini menggunakan Laptop Toshiba Satellite L645 dimana dalam laptop ini akan menjalankan berbagai software dan sistem operasi diantaranya sistem operasi Windows 8 Pro sebagai sistem operasi utama, software vmware player 10 menjalankan lima sistem operasi Windows xp dan satu sistem operasi Backtrack 5R3. Dalam sistem operasi Backtrack 5R3 menjalankan beberapa software diantaranya adalah Arpd dan Honeyd. Untuk memudahkan dalam penulisan bab 3 ini, penulis membuat diagram alur rancangan penelitian dibawah ini :

Gambar 3.1 Diagram alur rancangan penelitian

3.2 ANALISIS

Berkaitan dengan perancangan jaringan lokal dan rancangan skenario uji coba serangan DDoS (Distributed Denial of Service), sangatlah berperan pemahaman tentang serangan DDoS (Distributed Denial of Service).

DDoS adalah suatu serangan Denial of Service yang menggunakan banyak host penyerang untuk menyerang satu buah host target dalam sebuah jaringan. Serangan Denial of Service bersifat “satu lawan satu”, sehingga dibutuhkan sebuah host yang kuat, baik itu dari kekuatan pemrosesan atau sistem operasinya. demi membanjiri lalu lintas klien yang valid untuk mengakses layanan jaringan pada server yang dijadikan target serangan. Serangan DDoS ini menggunakan teknik canggih dibandingkan dengan serangan Denial of Service, yakni dengan meningkatkan serangan beberapa kali dengan menggunakan beberapa buah komputer sekaligus, sehingga dapat mengakibatkan server atau keseluruhan segmen jaringan dapat menjadi tidak berguna sama sekali bagi klien. Dalam penelitian skripsi ini digunakan empat botnet (robot dan network) yang nantinya akan digunakan untuk menyerang sebuah web server di sistem operasi Windows xp (virtual vmware), Dari penyerangan DDoS tersebut akan diketahui secara langsung dampak yang dihasilkan pada web server dan untuk pendeteksian serangan akan digunakan Honeyd agar serangan-serangan DDoS tersebut terdeteksi, juga akan dibuat sebuah file log untuk mencatat serangan-serangan tersebut yang kemudian akan dibuat sebuah grafik dan tabel terhadap traffic normal dan traffic serangan.

3.3 RANCANGAN PENELITIAN

Pada tahap ini dilakukan rancangan penelitian terhadap konsep dan metode yang digunakan, serta pengumpulan data-data mengenai yang dibutuhkan seperti paper, jurnal, makalah, buku, artikel, implementasi, dan sebagainya.

3.3.1 DEFINISI KEBUTUHAN SISTEM

Pada tahap ini dilakukan pendefinisian terhadap apa saja yang dibutuhkan untuk membangun tugas akhir ini, antara lain :

1. Kebutuhan Hardware

   Dalam pengerjaan tugas akhir ini menggunakan sebuah laptop dengan spesifikasi sebagai berikut: Toshiba Satellite L645, Intel Core i3 2.40 Ghz, Memory RAM 8 GB, HDD 320 GB.

2. Kebutuhan Software

   Software atau perangkat lunak yang digunakan dalam tugas akhir ini oleh attacker sebagai berikut:

   a. Windows 8 Pro

   Digunakan sebagai sistem operasi utama yang dapat menjalankan virtual machine vmware

   b. Mirc

   Digunakan untuk internet relay chat atau percakapan daring yang ber

   operasi di sistem operasi windows dan untuk memanggil serta memberi perintah penyerangan untuk botnet.

c. Vmware Player

Merupakan software yang digunakan untuk virtual machine (mesin virtual) fungsinya adalah untuk menjalankan banyak sistem operasi dalam satu perangkat keras (tentu saja perlu di perhatikan spesifikasi komputer yang digunakan) dan untuk menjalankan aplikasi yang ditunjuk untuk sistem operasi lainnya.

d. Backtrack 5 R3

Adalah sebuah sistem operasi yang berjalan pada virtual machine dan digunakan untuk mengemulasi virtual Honeyd.

e. Windows XP

Adalah sebuah sistem operasi yang berjalan pada virtual machine yang digunakan bersama Xampp untuk web lokal dan untuk menjalankan script zombie pada web browser.

f. Xampp

Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl

g. Honeyd

Merupakan Honeypot open source yang digunakan untuk mendeteksi serangan-serangan yang dilakukan oleh Attacker, dengan memberi konfigurasi-konfigurasi yang dapat menjebak cracker atau hacker.

h. Arpd

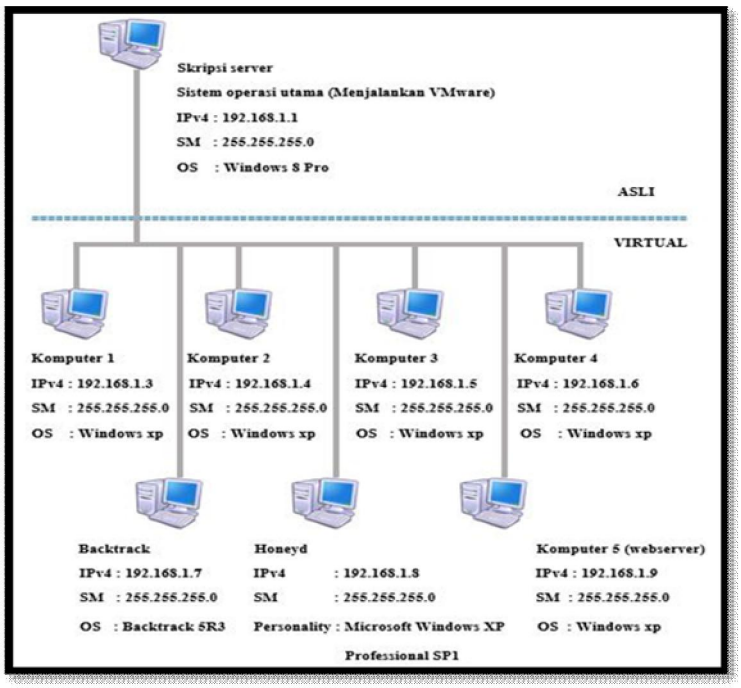
Merupakan software penunjang untuk Honeyd yang fungsinya sebagai daemon yang mendengarkan request ARP dan menjawab alamat IP yang tidak terpakai.

i. Wireshark

Merupakan salah satu aplikasi network analyzer (penganalisa jaringan) yang digunakan untuk menangkap paket-paket data dalam jaringan yang silih berganti.

3.3.2 RANCANGAN JARINGAN

Rancangan jaringan untuk penelitian skripsi ini adalah seperti gambar dibawah ini :



Gambar 3.2 Rancangan Jaringan Lokal

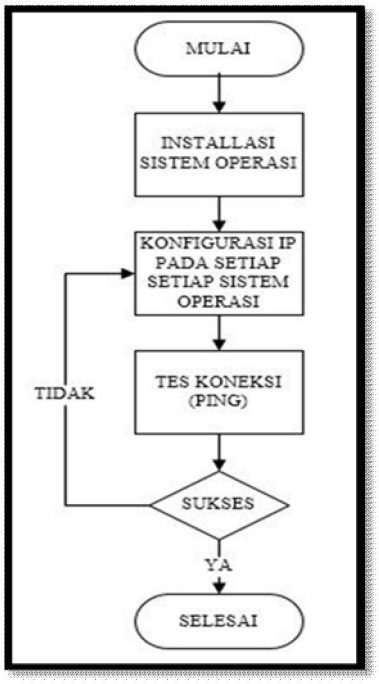
Seperti yang tertera pada gambar 3.2 secara sederhana menjelaskan dimana komputer dengan nama skripsi server dengan sistem operasi utama Windows 8

Pro dengan IPv4 192.168.1.1 dan subnet mask 255.255.255.0, tipe jaringan yang digunakan adalah network adapter host only, dimana komputer-komputer virtual terhubung dengan komputer sungguhan. Didalamnya juga ada berbagai sistem operasi diantaranya 5 sistem operasi Windows xp dan 1 sistem operasi Backtrack 5R3 dengan IP class C, diantaranya :

1. Komputer 1 dengan sistem operasi windows xp, IPv4 192.168.1.3, subnet mask 255.255.255.0
2. Komputer 2 dengan sistem operasi windows xp, IPv4 192.168.1.4, subnet mask 255.255.255.0
3. Komputer 3 dengan sistem operasi windows xp, IPv4 192.168.1.5, subnet mask 255.255.255.0
4. Komputer 4 dengan sistem operasi windows xp, IPv4 192.168.1.6, subnet mask 255.255.255.0
5. Sistem operasi backtrack 5R3 menggunakan IPv4 192.168.1.7, subnet mask 255.255.255.0
6. Komputer 5 dengan sistem operasi windows xp, IPv4 192.168.1.9, subnet mask 255.255.255.0
7. Honeyd dengan IPv4 192.168.1.8 (Honeyd sudah ada pada sistem operasi Backtrack 5R3 jadi dalam 1 sistem operasi, disini cuma membedakan IPv4 saja) untuk informasi Honeyd bisa menyamar sebagai sistem operasi ataupun Router untuk penelitian skripsi ini manyamar sebagai Windows xp.

Berdasarkan rancangan jaringan lokal tersebut untuk membuktikan bahwa semua sistem operasi baik pada sistem operasi asli dan sistem operasi pada virtual vmware tersambung satu sama lain akan dilakukan tes koneksi kepada semua

sistem operasi tersebut, berikut pada gambar 3.3 adalah diagram alur tes koneksi (ping) ke semua sistem operasi.

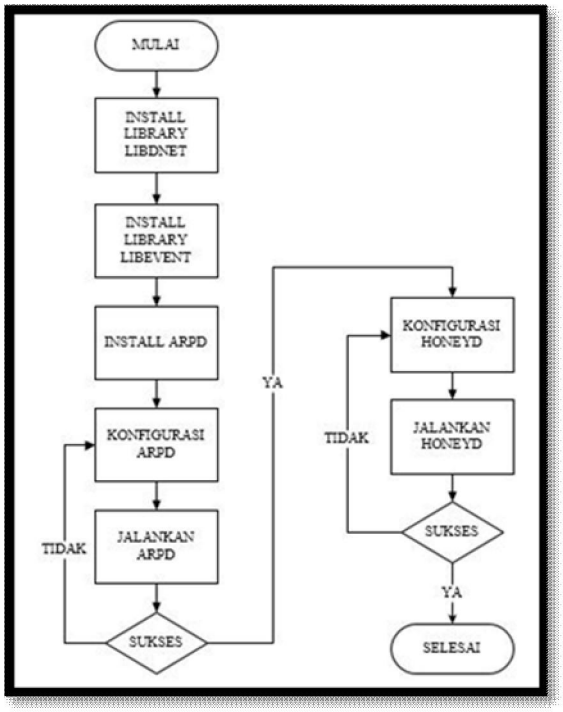


Gambar 3.3 Diagram alur tes koneksi (ping) ke semua sistem operasi

3.3.3 RANCANGAN HONEYPOT

Rancangan sistem Honeypot dalam skripsi ini memakai jenis Honeypot yaitu Honeyd. Honeyd merupakan Honeypot open source yang digunakan untuk mendeteksi serangan-serangan yang dilakukan oleh Attacker, dengan memberi konfigurasi-konfigurasi yang dapat menjebak cracker atau hacker, dalam penelitian skripsi ini menggunkan Honeyd untuk mendeteksi dan menganalisa serangan. Pada sistem operasi Backtrack 5R3 sudah tersedia Honeyd, untuk menjalankannya membutuhkan beberapa software pendukung seperti Arpd, Arpd berfungsi untuk mendengarkan paket ARP dan juga menjawab alamat IP yang tidak terpakai. Untuk menjalankan Arpd dibutuhkan beberapa library-library pendukung seperti Libdnet dan Libevent. Tanpa dua library tersebut Arpd tidak

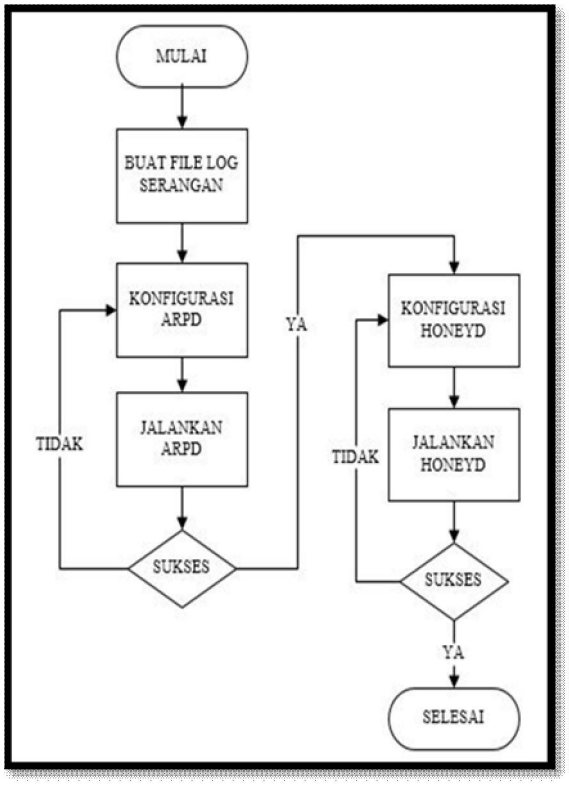
bisa di install. Pada Honeyd terdapat file konfigurasi yaitu Honeyd.conf, semua konfigurasi ada pada file ini. Didalamnya terdapat beberapa konfigurasi diantaranya identitas (personality), personality merupakan sebuah konfigurasi yang membuat Honeyd menyamar sebagai sistem operasi tertentu dan router, sesuai dengan pemberian nama pada personality tersebut dan ketika device lain terkoneksi dengan honeyd ini maka Honeyd akan dikenali. Port-port yang akan dibuka, jenis protokol-protokol yang di ijinkan dan yang akan diblokir dan yang terakhir adalah bind, bind digunakan untuk melakukan pemberian IP untuk setiap virtual Honeypot. Berikut ini adalah diagram alur rancangan implementasi Honeyd :



Gambar 3.4 Diagram alur rancangan implementasi Honeyd

Seperti yang tertera pada diagram alur diatas secara sederhana menjelaskan bahwa sebelum menjalankan Honeyd terlebih dahulu menginstall library-library yang dibutuhkan diantaranya adalah library libdnet, library libevent, dan arpd. Setelah semua itu terinstall di sistem operasi backtrack 5 R3 barulah

mengkonfigurasi arpd dan menjalankannya. Jika proses konfigurasi dan menjalankannya menemukan kendala atau tidak sukses maka akan kembali lagi ke proses konfigurasi arpd, dan jika sukses maka ke proses selanjutnya yaitu konfigurasi Honeyd dan menjalankan Honeyd. Jika menjalankan Honeyd menemukan kendala atau tidak sukses maka akan kembali lagi ke proses konfigurasi Honeyd dan jika sukses maka selesai dan Honeyd dapat digunakan.



Gambar 3.5 Rancangan implementasi log serangan dan Honeyd

Seperti yang tertera pada diagram alur diatas secara sederhana menjelaskan bahwa sebelum menjalankan arpd dan Honeyd terlebih dahulu untuk membuat file kosong atau file log serangan yang nantinya akan mencatat seluruh serangan yang telah dilakukan oleh Attacker, setelah membuat file log langkah berikutnya adalah konfigurasi arpd dan menjalankannya jika dalam menjalankan arpd menemukan kendala atau tidak sukses maka akan kembali ke proses konfigurasi arpd, jika sukses maka akan ke proses konfigurasi Honeyd dan menjalankannya. jika dalam

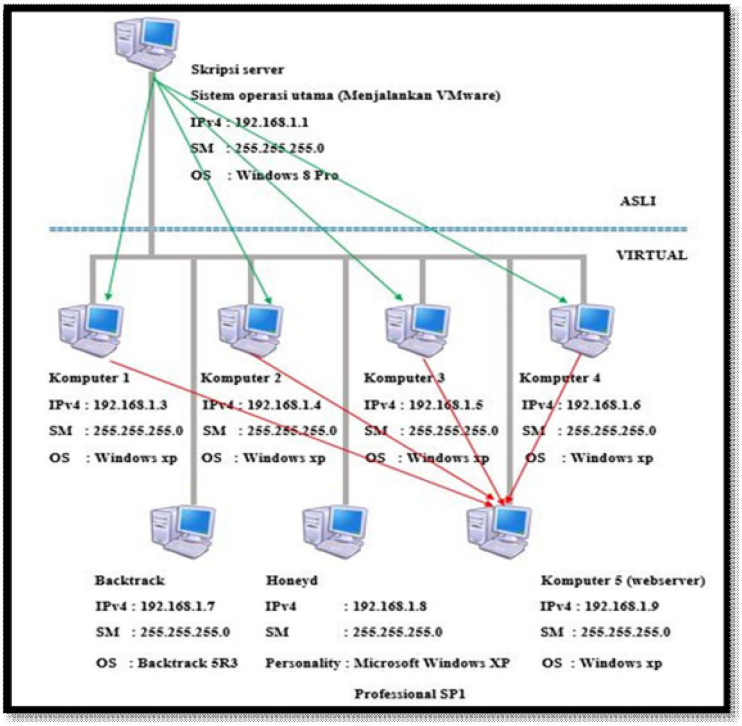
menjalankan Honeyd menemukan kendala atau tidak sukses maka akan kembali lagi ke proses konfigurasi Honeyd dan jika sukses maka seluruh proses implementasi telah selesai.

3.4 SKENARIO UJI COBA DAN PENJELASAN

Skenario uji coba dan penjelasan dalam skripsi ini dilakukan skenario uji coba sebanyak 2 kali. Skenario pertama penyerangan dilakukan ke web server pada komputer 5 sistem operasi Windows xp di virtual vmware dengan jenis serangan HTTP Flood dan SYN Flood, hal ini untuk membuktikan bahwa serangan DDoS yang telah dilakukan oleh Attacker berjalan dengan benar. Untuk dampak yang dihasilkan dari serangan ini, web server pada komputer 5 sistem operasi windows xp di virtual vmware akan meresponnya.

Skenario kedua penyerangan dilakukan ke Honeyd dengan jenis serangan HTTP Flood dan SYN Flood, hal ini untuk mendeteksi dan menganalisa serangan-serangan DDoS, juga akan dibuat file log untuk mencatat serangan-serangan tersebut, pada file log tersebut nantinya akan diolah di log analyzer. sebagai hasilnya akan dibuatkan sebuah grafik dan tabel, untuk memudahkan dalam menganalisa serangan-serangan yang telah dilakukan oleh Attacker, sehingga bisa membandingkan antara traffic normal dan traffic serangan. Pada traffic normal disini diasumsikan bahwa User yang melakukan akses ke web emulasi Honeyd, sedangkan traffic serangan disini diasumsikan bahwa Attacker yang menyerang web server emulasi Honeyd tersebut. Untuk lebih memahami pola rancangan skenario pertama akan dibahas pada sub bab 3.4.1, sedangkan skenario kedua akan dibahas pada sub bab 3.4.2.

3.4.1 Skenario Pertama



Gambar 3.6 Rancangan serangan HTTP Flood dan SYN Flood ke web server

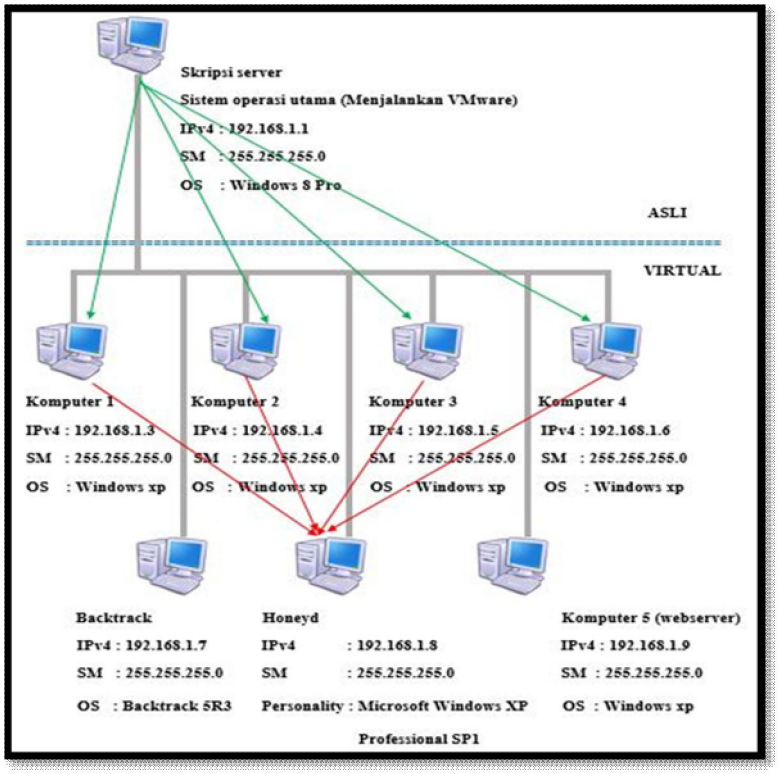
Seperti yang tertera pada gambar 3.6 secara sederhana menjalaskan bahwa skripsi server dengan IPv4 192.168.1.1 SM 255.255.255.0 menjalankan aplikasi Mirc (command and control) server secara lokal. Kemudian juga menjalankan beberapa sistem operasi di virtual vmware, diantaranya :

1. Komputer 1 dengan IPv4 192.168.1.3 SM 255.255.255.0
2. Komputer 2 dengan IPv4 192.168.1.4 SM 255.255.255.0
3. Komputer 3 dengan IPv4 192.168.1.5 SM 255.255.255.0
4. Komputer 4 dengan IPv4 192.168.1.6 SM 255.255.255.0
5. Komputer 5 (web server) dengan IPV4 192.168.1.9 SM 255.255.255.0

Ke-empat sistem operasi di virtual vmware tersebut yakni komputer 1 sampai komputer 4 menjalankan aplikasi Xampp untuk mengaktifkan server lokal Apache lalu menjalankan script zombie melalui pemanggilan pada web browser.

Pada skripsi server yang menjalankan Mirc (command and control) server akan tampil keempat botnet (robot and network) dan kemudian diperintah untuk menyerang komputer 5 (web server). Dalam skenario serangan akan ada perbedaan sebelum dan saat terjadinya serangan, ketika user lain melakukan akses ke web server tersebut sebelum serangan dan saat terjadi serangan maka hasilnya juga pastinya akan berbeda. Setelah membahas rancangan serangan HTTP Flood dan SYN Flood pada web server pada komputer 5 sistem operasi windows xp di virtual vmware berikutnya akan dibahas rancangan serangan HTTP Flood dan SYN Flood pada Honeyd.

3.4.2 Skenario Kedua



Gambar 3.7 Rancangan serangan HTTP Flood dan SYN Flood ke Honeyd

Seperti yang tertera pada gambar 3.7 secara sederhana menjalaskan bahwa skripsi server dengan IPv4 192.168.1.1 SM 255.255.255.0 menjalankan aplikasi

Mirc (command and control) server secara lokal. Kemudian juga menjalankan beberapa sistem operasi di Vmware Player, diantaranya :

1. Komputer 1 dengan IPv4 192.168.1.3 SM 255.255.255.0
2. Komputer 2 dengan IPv4 192.168.1.4 SM 255.255.255.0
3. Komputer 3 dengan IPv4 192.168.1.5 SM 255.255.255.0
4. Komputer 4 dengan IPv4 192.168.1.6 SM 255.255.255.0
5. Backtrack 5R3 dengan IPv4 192.168.1.7 SM 255.255.255.0
6. Honeyd dengan IPv4 192.168.1.8 SM 255.255.255.0

Ke-empat sistem operasi tersebut yakni komputer 1 sampai komputer 4 menjalankan aplikasi Xampp untuk mengaktifkan server lokal Apache lalu menjalankan script zombie melalui pemanggilan pada web browser. Di skripsi server yang menjalankan Mirc (command and control) server akan tampil keempat botnet (robot and network) dan kemudian diperintah untuk menyerang Honeyd yang berada pada sistem operasi Backtrack 5R3, untuk mendokumentasikan hasil dari serangan-serangan tersebut dibuat file log yang bertujuan untuk mencatat hasil dari traffic normal dan serangan-serangan DDoS tersebut, yang nantinya pada file log tersebut akan diolah pada log analyzer untuk memudahkan dalam memahaminya.

3.5 ANALISA DAN PEMBUKTIAN SERANGAN

Dalam pendeteksian dan analisa dibutuhkan berbagai referensi seperti jurnal, artikel, paper dll, agar pendeteksian serangan dan analisa yang dilakukan lebih akurat. Berikut adalah referensi-referensinya :

Ada berbagai macam serangan DDoS dan ada dua kelas utama dari serangan DDoS yaitu berupa bandwidth depletion attack (serangan menghabiskan

bandwidth) dan resource depletion attack (serangan menghabiskan sumber daya). Bandwidth depletion attack dirancang untuk membanjiri korban dengan lalu lintas yang tidak diinginkan sedangkan resource depletion attack dirancang untuk membanjiri sistem target korban, target utamanya adalah proses dari mesin target. (Stephen M.Specht, Ruby B.Lee : 2)

Fitur mencolok dari serangan SYN Flood adalah bahwa penyerang mengirimkan sejumlah besar TCP SYN paket permintaan dengan sumber alamat IP palsu. Hal ini menyebabkan sisi server mengkonsumsi sejumlah besar sumber daya. untuk mempertahankan daftar yang sangat besar koneksi setengah terbuka, akhirnya mengarah ke server dan kehabisan sumber daya dan menjadi tidak mampu memberikan layanan normal. (NSFOCUS: 1)

Serangan HTTP Flood akan membangun koneksi TCP yang normal ke server dan terus-menerus mengirimkan banyak pemanggilan yang secara dramatis mengkonsumsi sumber daya. Sebuah khas HTTP GET adalah penyerang mengirimkan sejumlah besar pemanggilan untuk host server yang mengkonsumsi sumber daya server hanya dalam beberapa menit. (NSFOCUS: 5)

Setelah mengetahui penjelasan dan referensi tentang HTTP Flood dan SYN Flood dari berbagai sumber yang sudah dicantumkan, sekarang saatnya untuk memaparkan parameter-parameter keberhasilan sesuai dengan referensi-referensi yang sudah dijelaskan, dengan parameter-parameter keberhasilan tersebut diharapkan mampu untuk melakukan proses analisa. Parameter-parameter keberhasilan tersebut adalah :

1. Jumlah paket yang diterima oleh Honeyd sangat banyak.
2. Besar paket yang diterima oleh Honeyd sangat besar.

3. Dilakukan oleh banyak alamat IP yang digunakan Attacker untuk menyerang korban.
4. Terjadinya perbedaan waktu yang pendek saat Attacker melakukan penyerangan

Parameter-parameter keberhasilan tersebut akan digunakan untuk proses analisa terhadap serangan DDoS HTTP Flood dan SYN Flood.

3.5.1 Rancangan Analisa Serangan DDoS HTTP Flood

Berikut akan dipaparkan langkah-langkah proses analisa serangan DDoS HTTP Flood pada Honeyd:

1. User melakukan akses normal ke web yang diemulasi Honeyd.
2. File log Honeyd akan mencatat interaksi yang ditujukan padanya.
3. Setelah melakukan akses normal yang dilakukan oleh User, Attacker melakukan serangan DDoS HTTP Flood.
4. File log Honeyd akan mencatat interaksi lagi yang dtujukan padanya.
5. Setelah melakukan akses normal dan serangan DDoS HTTP Flood ke web emulasi Honeyd. Maka untuk memahaminya File log yang dihasilkan akan diolah dan disamakan dengan parameter keberhasilan yang sudah dibuat.
6. Langkah terakhir membuat kesimpulan sesuai dengan proses analisa.

Langkah-langkah tersebut akan dilakukan sebanyak 20 kali percobaan agar data yang diperoleh cukup untuk melakukan proses analisa. Setelah mengetahui rancangan analisa serangan DDoS HTTP Flood, berikutnya pada sub bab 3.5.2 akan dijelaskan rancangan analisa serangan DDoS SYN Flood.

3.5.2 Rancangan Analisa Serangan DDoS SYN Flood

Berikut akan dipaparkan langkah-langkah proses analisa serangan DDoS SYN Flood pada Honeyd :

1. User melakukan akses normal ke web yang diemulasi Honeyd.
2. File log Honeyd akan mencatat interaksi yang ditujukan padanya.
3. Setelah melakukan akses normal yang dilakukan oleh User, Attacker melakukan serangan DDoS SYN Flood.
4. File log Honeyd akan mencatat interaksi lagi yang dtujukan padanya.
5. Setelah melakukan akses normal dan serangan DDoS SYN Flood ke web emulasi Honeyd. Maka untuk memahaminya File log yang dihasilkan akan diolah dan disamakan dengan parameter keberhasilan yang sudah dibuat.
6. Langkah terakhir membuat kesimpulan sesuai dengan proses analisa.

Langkah-langkah tersebut akan dilakukan sebanyak 20 kali percobaan agar data yang diperoleh cukup untuk melakukan proses analisa.

BAB IV

HASIL DAN PEMBAHASAN

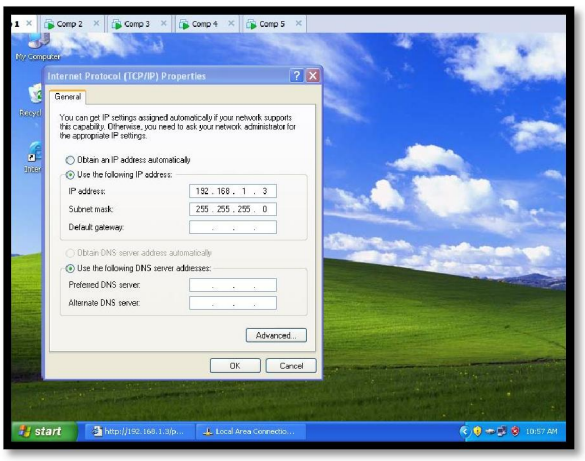
Pada bab ini akan membahas mengenai proses installasi program dan konfigurasinya, Honeyd serta library-library pendukung. berbagai implementasi dari skenario serangan, pendeteksian dan analisa serangan.

4.1 IMPLEMENTASI

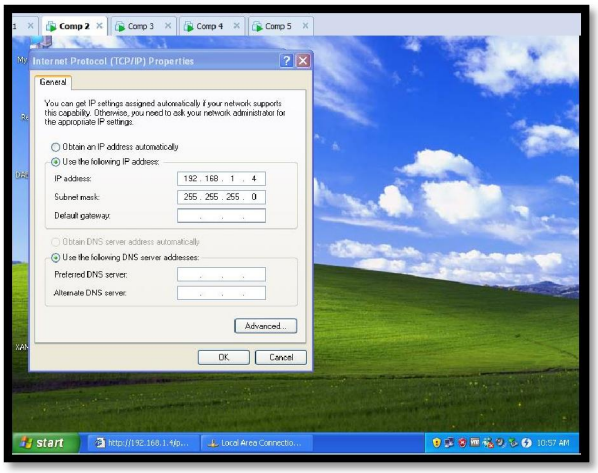
Implementasi pada bab ini akan dilakukan berdasarkan pada rancangan penelitian pada bab III, berikut adalah hasil implementasi dari rancangan penelitian tersebut.

4.1.1 Konfigurasi IP Dari Setiap OS Di Virtual

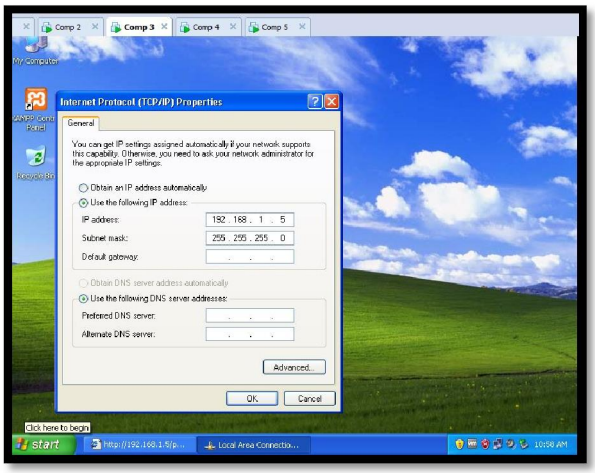
Konfigurasi IP dari setiap sistem operasi di virtual vmware merupakan hal yang harus dilakukan karena untuk setiap sistem operasi pasti memiliki IP yang berbeda-beda, berikut adalah screenshoot dari setiap konfigurasi IP pada os di virtual.



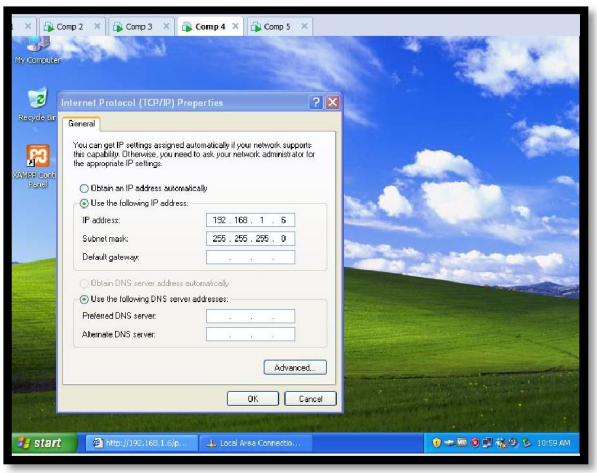
Gambar 4.1 Konfigurasi IP pada komputer 1 di virtual vmware



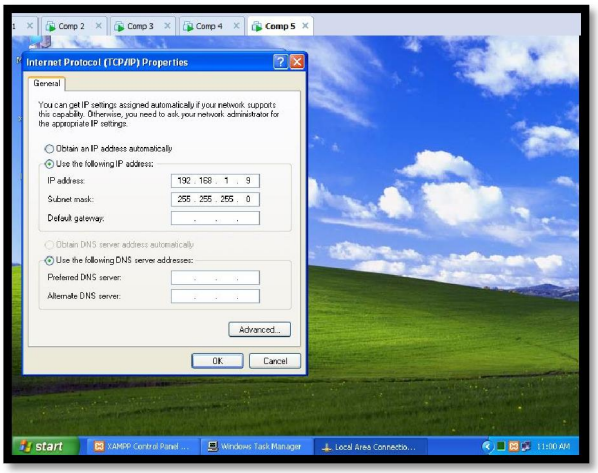
Gambar 4.2 Konfigurasi IP pada komputer 2 di virtual vmware



Gambar 4.3 Konfigurasi IP pada komputer 3 di virtual vmware



Gambar 4.4 Konfigurasi IP pada komputer 4 di virtual vmware



Gambar 4.5 Konfigurasi IP pada komputer 5 di virtual vmware

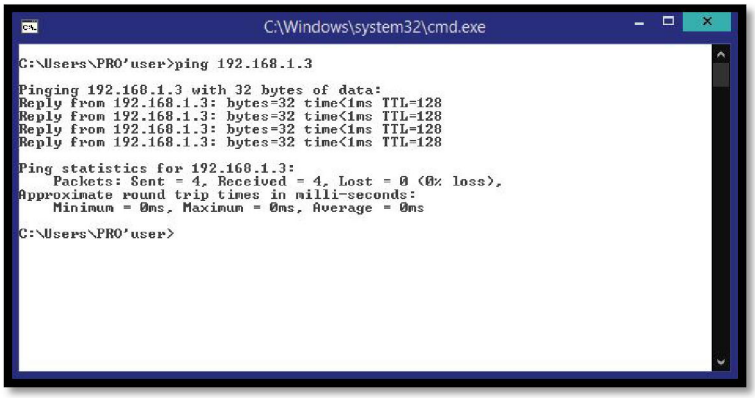
Gambar 4.6 Konfigurasi IP pada os backtrack 5R3

Gambar-gambar diatas mulai dari gambar 4.1-4.6 merupakan konfigurasi alamat IP pada setiap os di virtual vmware, IP dari os virtual tersebut diantaranya:

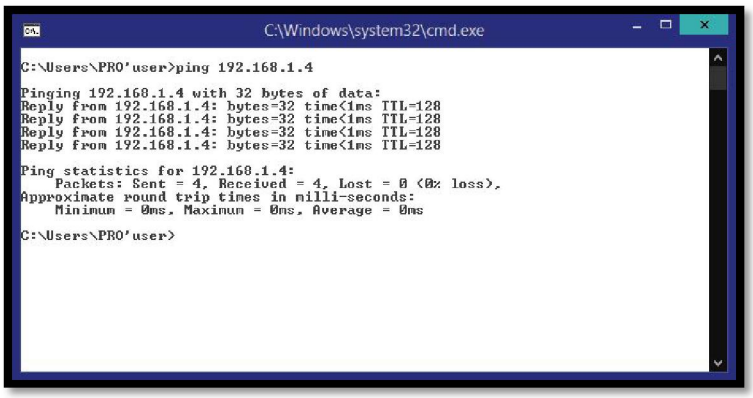
- Komputer 1 : 192.168.1.3
- Komputer 2 : 192.168.1.4
- Komputer 3 : 192.168.1.5
- Komputer 4 : 192.168.1.6
- Komputer 5 : 192.168.1.9
- Backtrack 5R3 : 192.168.1.7

IP (Internet Protokol) dari setiap komputer 1, komputer 2, komputer 3, komputer 4, komputer 5 dan backtrack menggunakan IP Class C. Berikut pada sub bab 4.1.2 akan dilakukan tes koneksi dari os asli ke os virtual.

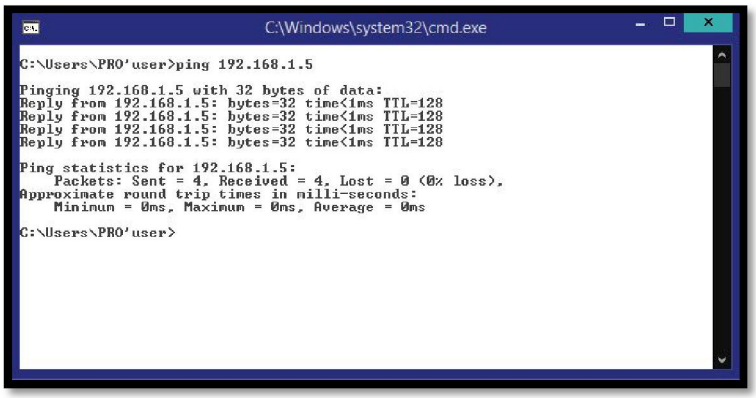
4.1.2 Tes Koneksi Dari OS Asli Ke Virtual



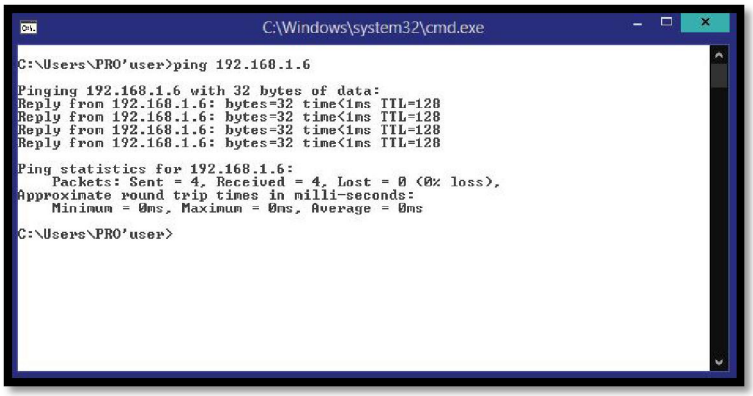
Gambar 4.7 Tes koneksi dari os asli 192.168.1.1 ke os virtual 192.168.1.3



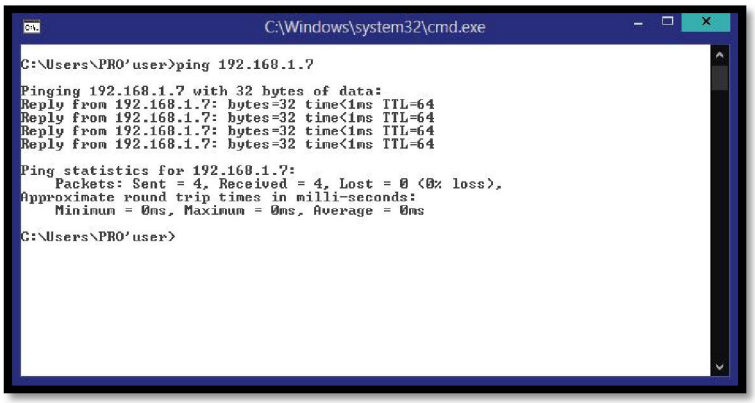
Gambar 4.8 Tes Koneksi dari os asli 192.168.1.1 ke os virtual 192.168.1.4



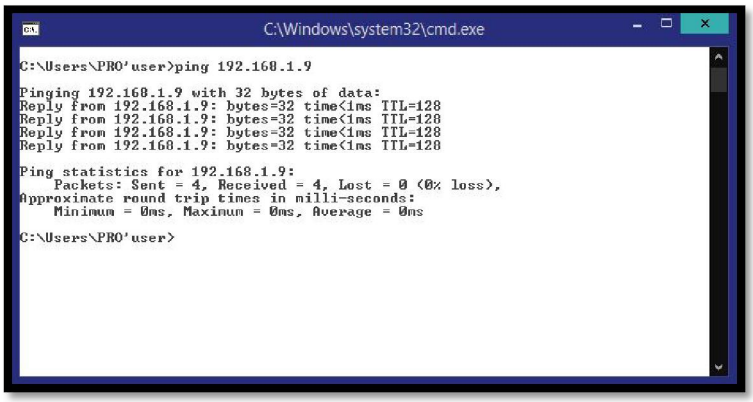
Gambar 4.9 Tes koneksi dari os asli 192.168.1.1 ke os virtual 192.168.1.5



Gambar 4.10 Tes koneksi dari os asli 192.168.1.1 ke os virtual 192.168.1.6



Gambar 4.11 Tes koneksi dari os asli 192.168.1.1 ke os baktrack 192.168.1.7



Gambar 4.12 Tes koneksi dari os asli 192.168.1.1 ke os virtual 192.168.1.9

Gambar-gambar diatas mulai dari gambar 4.7-4.12 merupakan hasil tes koneksi (ping) dari os asli 192.168.1.1 ke os virtual mulai dari :

- 192.168.1.1 ke 192.168.1.3 = Terhubung
- 192.168.1.1 ke 192.168.1.4 = Terhubung

- 192.168.1.1 ke 192.168.1.5 = Terhubung
- 192.168.1.1 ke 192.168.1.6 = Terhubung
- 192.168.1.1 ke 192.168.1.7 = Terhubung
- 192.168.1.1 ke 192.168.1.9 = Terhubung

4.1.3 Installasi Arpd Dan Library-library pendukung

Sebelum menjelaskan tentang installasi Honeyd ada baiknya mengenal Honeyd terlebih dahulu, Honeyd merupakan honeypot open source yang digunakan untuk mendeteksi serangan-serangan yang dilakukan oleh attacker, dengan memberi konfigurasi-konfigurasi yang dapat menjebak cracker/hacker.

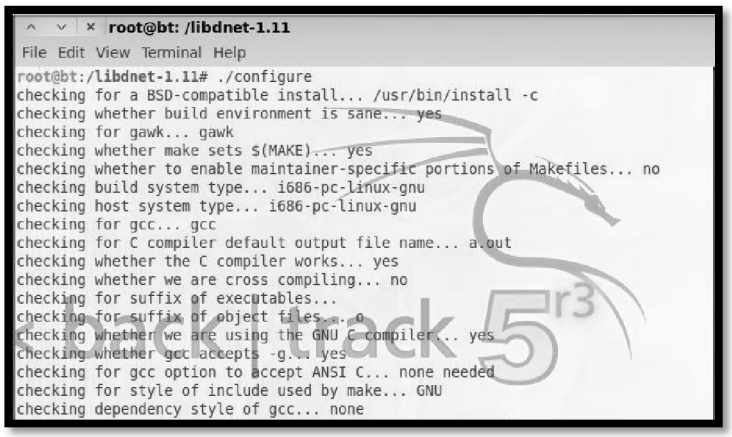
Sebelum memulai proses installasi Honeyd, ada beberapa library-library dan aplikasi yang harus diinstal terlebih dahulu, berikut diantaranya :

1. Library Libdnet :

Library libdnet merupakan suatu library yang berada pada linux yang dibutuhkan untuk menjalankan aplikasi Arpd. Dibawah ini merupakan hasil screenshoot dari proses installasi library libdnet-1.11

Gambar 4.13 Proses ekstrak file libdnet-1.11.tar.gz

Berdasarkan gambar 4.13 bahwa perintah untuk ekstrak file libdnet-1.11 dengan format file tar.gz adalah dengan perintah tar –xvf libdnet-1.11.tar.gz. setelah menjalankan perintah tersebut maka proses ekstrak file akan segera dimulai dan akan membuat sebuah folder libdnet-1.11 di direktori Backtrack 5R3. jika sudah selesai proses ekstrak file maka akan ke proses selanjutnya :



Gambar 4.14 Proses cek file pada direktori libdnet-1.11

Berdasarkan gambar 4.14 bahwa perintah proses untuk cek file pada direktori libdnet-1.11 adalah dengan perintah ./configure. Setelah menjalankan perintah tersebut maka proses cek file akan segera dimulai dan akan mengecek file-file apa saja yang dibutuhkan untuk libdnet-1.11. jika sudah selesai proses cek file pada libdnet-1.11 maka akan ke proses selanjutnya.

Gambar 4.15 Proses membuat file install pada direktori libdnet-1.11

Berdasarkan gambar 4.15 bahwa perintah untuk proses membuat file install pada direktori libdnet-1.11 adalah dengan perintah make. setelah menjalankan perintah tersebut maka proses membuat file install akan segera dimulai sehingga dapat menghasilkan file installer. Jika sudah selesai proses membuat file install maka akan ke proses selanjutnya.

Gsmbar 4.16 Proses installasi libdnet-1.11

Berdasarkan gambar 4.16 bahwa perintah untuk proses installasi libdnet-1.11 adalah dengan menggunakan perintah make install. Setelah menjalankan perintah tersebut maka proses membuat installasi libdnet-1.11 akan segera dimulai dan akan melakukan proses install ke direktori backtrack 5R3.

2. Library Libevent

Library libevent adalah library yang berada pada linux merupakan software library yang menyediakan pemberitahuan secara asynchronous. Api (Application Programming Interfaces) pada library libevent menyediakan mekanisme untuk mengeksekusi fungsi callback ketika peristiwa tertentu pada file descriptor atau setelah batas waktu telah tercapai. Library libevent dan library libdnet adalah

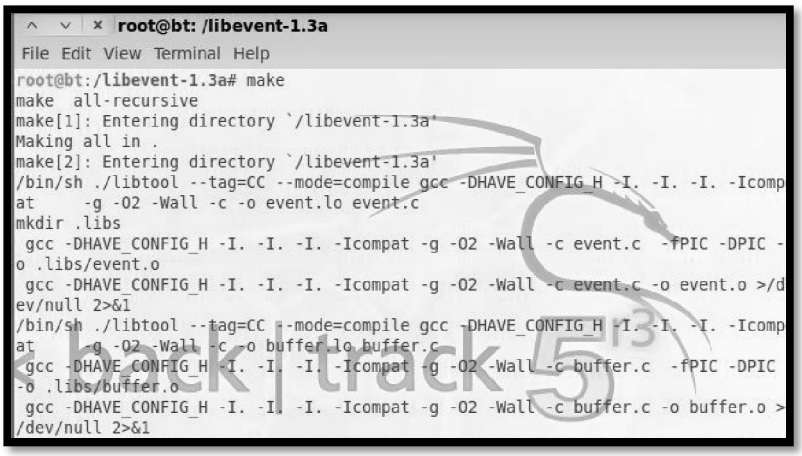
library yang dibutuhkan untuk installasi arpd. Dibawah ini merupakan hasil screenshoot dari proses installasi library libevent :

Gambar 4.17 Proses ekstrak file libevent-1.3a.tar.gz

Berdasarkan gambar 4.17 bahwa perintah untuk ekstrak file libevent-1.3a dengan format file tar.gz adalah dengan perintah tar –xvf libevent-1.3a.tar.gz. setelah menjalankan perintah tersebut maka proses ekstrak file akan segera dimulai dan akan membuat sebuah folder libevent-1.3a di direktori Backtrack 5R3. jika sudah selesai proses ekstrak file maka akan ke proses selanjutnya.

Gambar 4.18 Proses cek file pada direktori libevent-1.3a

Berdasarkan gambar 4.18 bahwa perintah proses untuk cek file pada direktori libevent-1.3a adalah dengan perintah ./configure. Setelah menjalankan perintah tersebut maka proses cek file akan segera dimulai dan akan mengecek file-file apa saja yang dibutuhkan untuk libevent-1.3a. jika sudah selesai proses cek file pada libevent-1.3a maka akan ke proses selanjutnya.



Gambar 4.19 Proses membuat file install pada direktori libevent-1.3a

Berdasarkan gambar 4.19 bahwa perintah untuk proses membuat file install pada direktori libevent-1.3a adalah dengan perintah make. setelah menjalankan perintah tersebut maka proses membuat file install akan segera dimulai sehingga dapat menghasilkan file installer. Jika sudah selesai proses membuat file install maka akan ke proses selanjutnya.

Gambar 4.20 Proses installasi libevent-1.3a

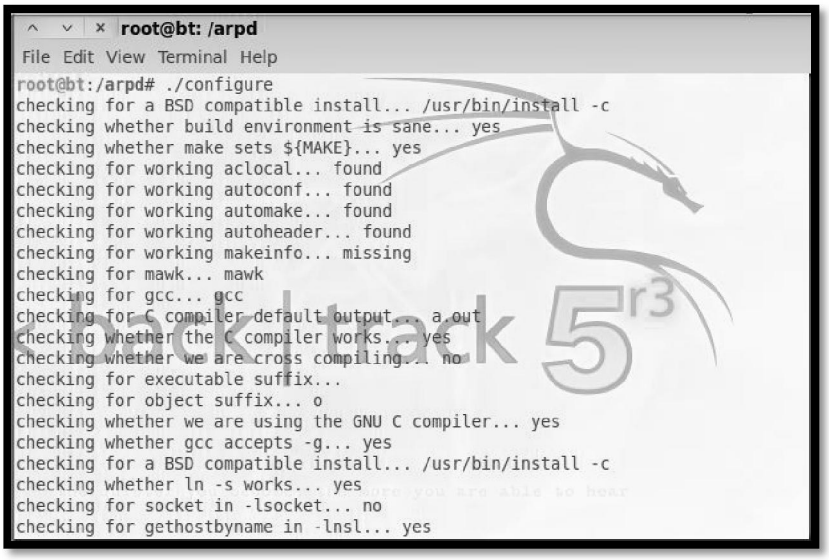
Berdasarkan gambar 4.20 bahwa perintah untuk proses installasi libevent-1.3a adalah dengan menggunakan perintah make install. Setelah menjalankan perintah tersebut maka proses membuat installasi libevent-1.3a akan segera dimulai dan akan melakukan proses install ke direktori backtrack 5R3.

3. Install Aplikasi Arpd

Aplikasi Arpd adalah daemon yang mendengarkan permintaan ARP dan jawaban untuk alamat IP yang tidak terpakai/terisi. Arpd sendiri merupakan aplikasi yang dijalankan bersama Honeyd. Dibawah ini merupakan hasil screenshoot dari proses installasi Aplikasi Arpd :

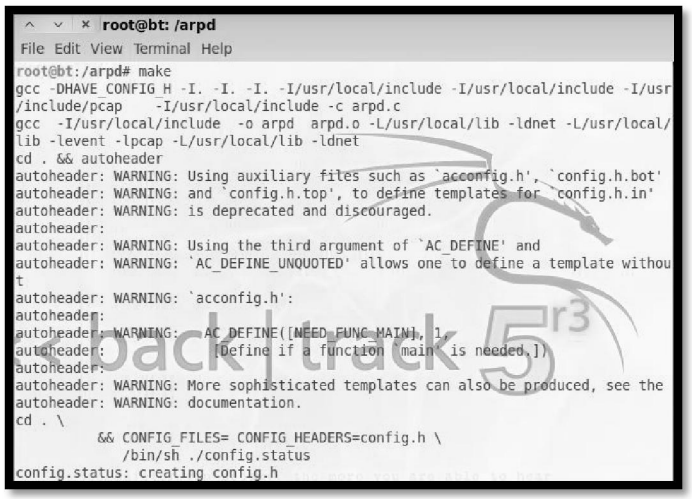
Gambar 4.21 Proses ekstrak file aplikasi arpd-0.2.tar.gz

Berdasarkan gambar 4.21 bahwa perintah untuk ekstrak file aplikasi arpd dengan format file tar.gz adalah dengan perintah tar –xvf arpd-0.2.tar.gz. setelah menjalankan perintah tersebut maka proses ekstrak file akan segera dimulai dan akan membuat sebuah folder arpd di direktori Backtrack 5R3. jika sudah selesai proses ekstrak file maka akan ke proses selanjutnya.



Gambar 4.22 proses cek file pada direktori arpd

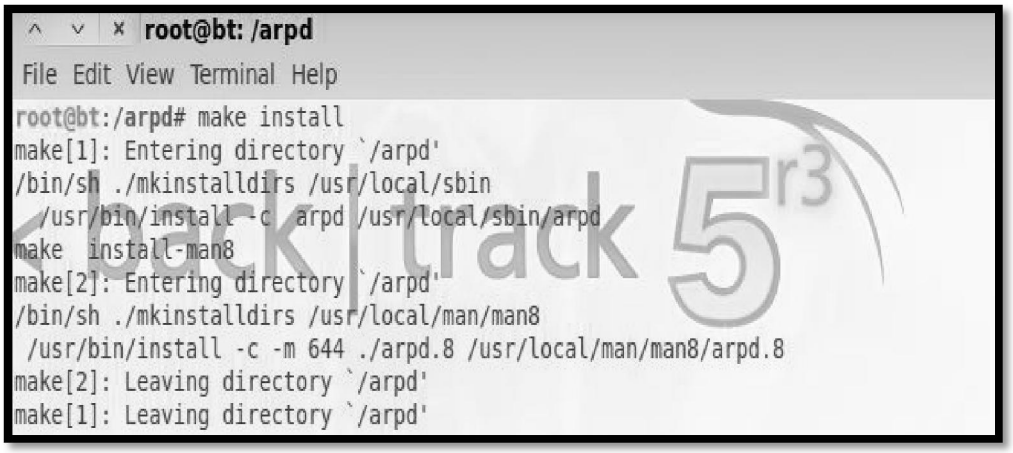
Berdasarkan gambar 4.22 bahwa perintah proses untuk cek file pada direktori arpd adalah dengan perintah ./configure. Setelah menjalankan perintah tersebut maka proses cek file akan segera dimulai dan akan mengecek file-file apa saja yang dibutuhkan untuk arpd. jika sudah selesai proses cek file pada arpd maka akan ke proses selanjutnya.



Gambar 4.23 Proses membuat file install pada direktori arpd

Berdasarkan gambar 4.23 bahwa perintah untuk proses membuat file install pada direktori arpd adalah dengan perintah make. setelah menjalankan perintah tersebut

maka proses membuat file install akan segera dimulai sehingga dapat menghasilkan file installer. Jika sudah selesai proses membuat file install maka akan ke proses selanjutnya.

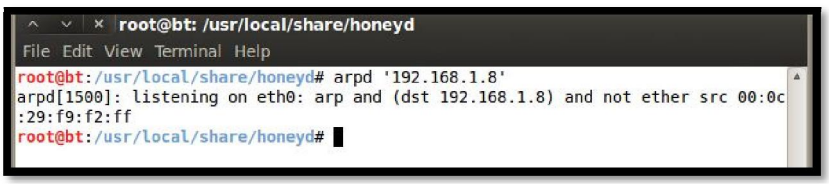


Gambar 4.24 Proses installasi arpd

Berdasarkan gambar 4.24 bahwa perintah untuk proses installasi arpd adalah dengan menggunakan perintah make install. Setelah menjalankan perintah tersebut maka proses membuat installasi arpd akan segera dimulai dan akan melakukan proses install ke direktori backtrack 5R3.

4.1.4 Konfigurasi Dan Menjalankan Arpd

Konfigurasi dan menjalankan arpd merupakan suatu hal yang harus dilakukan karena aplikasi arpd merupakan daemon yang mendengarkan permintaan ARP dan jawaban untuk alamat IP yang tidak terpakai/terisi.



Gambar 4.25 Perintah menjalankan arpd

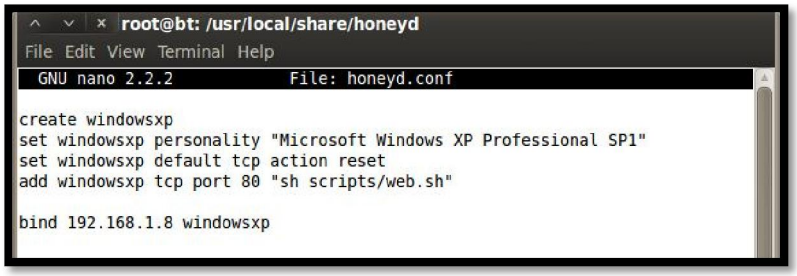
Berdasarkan gambar 4.25 bahwa setelah masuk ke direktori Honeyd dan berada di direktori tersebut maka proses selanjutnya adalah menjalankan aplikasi arpd dengan perintah : arpd '192.168.1.8'. agar paket IP yang dikirim ke IP 192.168.1.8 dapat dibalas oleh Honeyd sehingga seolah-olah merupakan komputer yang asli. ip tersebut harus sama dengan IP Honeyd, karena arpd dijalankan dengan Honeyd.

4.1.5 Konfigurasi Dan Menjalankan Honeyd

Setelah semua library-library libdnet, libevent dan aplikasi arpd sudah terinstall, sekarang saatnya untuk menjalankan Honeyd. Tapi hal yang harus diperhatikan adalah sebelum menjalankan Honeyd terlebih dahulu untuk menjalankan aplikasi arpd, berikut adalah screenshoot dan penjelasannya :

Gambar 4.26 Perintah untuk masuk ke direktori Honeyd

Berdasarkan gambar 4.26 bahwa untuk masuk ke direktori Honeyd terlebih dahulu untuk mengetikkan perintah pada terminal backtrack 5R3 sebagai berikut : cd /usr/local/share/honeyd. cd merupakan kepanjangan dari change directory. Setelah mengetikkan perintah tersebut maka akan dibawa ke direktori Honeyd.

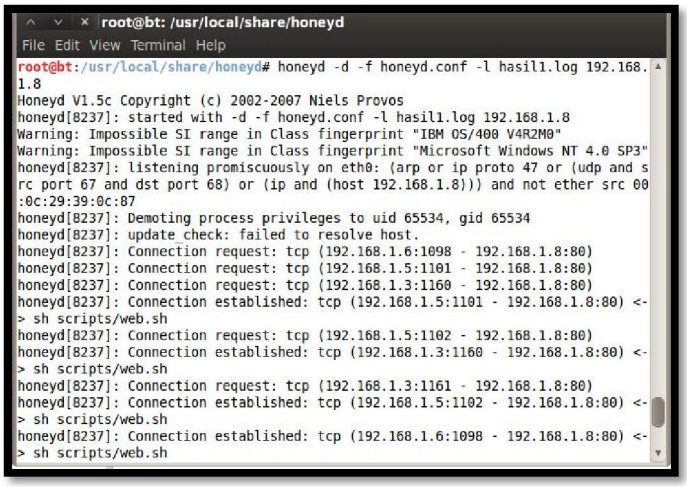


Gambar 4.27 Konfigurasi honeyd.conf

Berdasarkan gambar 4.27 untuk konfigurasi honeyd.conf terlebih dahulu masuk ke direktori Honeyd. Setelah itu mengetikkan perintah nano honeyd.conf.

1. create windowsxp = Memberi nama pada konfigurasi, nama tersebut bisa diisi dengan sesuai keinginan, nama tersebut juga berfungsi sebagai variabel yang dapat dipanggil
2. set windowsxp personality “Microsoft Windows XP Professional SP1” = Personality digunakan untuk mengadaptasi sistem operasi tertentu untuk mengelabuhi scanner fingerprint semacam Nmap dan ketika device lain terkoneksi dengan Honeyd ini maka akan dikenali sebagai Windows XP SP1.
3. set windowsxp default tcp action reset = Menyatakan secara default semua port TCP akan ditutup tetapi tetap memberikan alert. Selain reset ada kemungkinan lain untuk port TCP yaitu open dan block. Untuk open berarti semua port pada TCP akan dibuka dan dapat memberikan reply dan alert, sedangkan untuk block semua paket akan didrop dan tidak ada reply ataupun alert.
4. add windowsxp tcp port 80 “sh scripts/web.sh” = Menyatakan bahwa port 80 pada TCP dibuka. Meskipun script set windowsxp default tcp action reset menyatakan bahwa semua port TCP akan ditutup, tetapi dengan menambahkan script seperti add windowsxp tcp port 80 “sh scripts/web.sh membuat TCP port 80 akan terbuka dan memberi servis web server.

5. bind 192.168.1.8 windowsxp = Menyatakan bahwa IP address 192.168.1.8 digunakan oleh baris konfigurasi windowsxp dan ip tersebut digunakan sebagai ip Honeyd.



Gambar 4.28 Tampilan saat Honeyd dijalankan

Berdasarkan Gambar 4.28 adalah tampilan saat Honeyd dijalankan, perintah untuk menjalankannya adalah honeyd –d -f honeyd.conf –l hasil1.log 192.168.1.8. berikut adalah penjelasannya :

1. –d : Digunakan untuk menampilkan alert secara realtime, jika tidak menggunakan –d maka Honeyd akan berjalan secara background.
2. –f : Digunakan untuk mengambil file konfigurasi honeyd.conf. jika tidak menggunakan –f maka Honeyd akan berjalan dengan konfigurasi default.
3. –l : Digunakan untuk membuat file log. File log sendiri digunakan untuk mencatat interaksi apa saja yang masuk pada Honeyd. Dengan adanya file log ini maka serangan-serangan yang dilakukan oleh Attacker dapat didokumentasikan.

4.2 IMPLEMENTASI SKENARIO SERANGAN 1

Skenario serangan 1 bertujuan untuk membuktikan bahwa serangan DDoS HTTP Flood dan SYN Flood sudah dijalankan dengan benar ke web server korban, berikut pada sub bab 4.2.1 adalah penjelasan dari skenario serangannya

4.2.1 Skenario Serangan Pertama Ke Web server Komputer 5

Skenario serangan pertama dilakukan penyerangan ke web server dengan jenis serangan HTTP Flood. hal ini untuk membuktikan bahwa serangan DDoS tersebut sudah dijalankan dengan benar ke web server korban, juga untuk membuktikan sebelum serangan dan saat serangan dijalankan, web server pada komputer 5 sistem operasi windows xp di virtual vmware tersebut akan meresponnya.

1. Serangan DDoS HTTP Flood

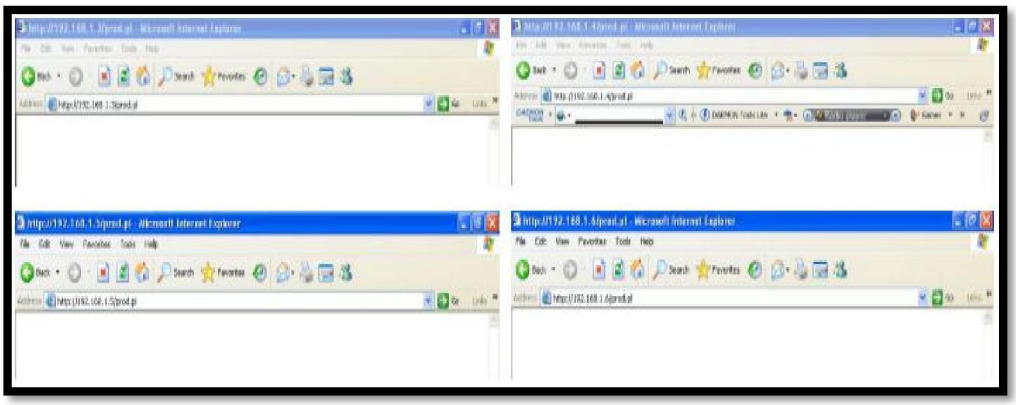
Nama Skenario : Serangan DDoS HTTP Flood pada web server
Korban pada komputer 5.

Tujuan : Skenario serangan ini bertujuan untuk
Membuktikan bahwa serangan HTTP Flood
telah dijalankan dengan benar.

Kebutuhan Awal : Web server pada komputer 5 sistem operasi
Windows xp di virtual Vmware harus bisa diakses. Script zombie harus dijalankan pada

masing-masing web browser, 4 botnet harus di load pada mirc.

Parameter Keberhasilan : Dikatakan berhasil saat dilakukan serangan HTTP Flood kondisi cpu usage pada sistem operasi windows xp yang digunakan sebagai web server meningkat.

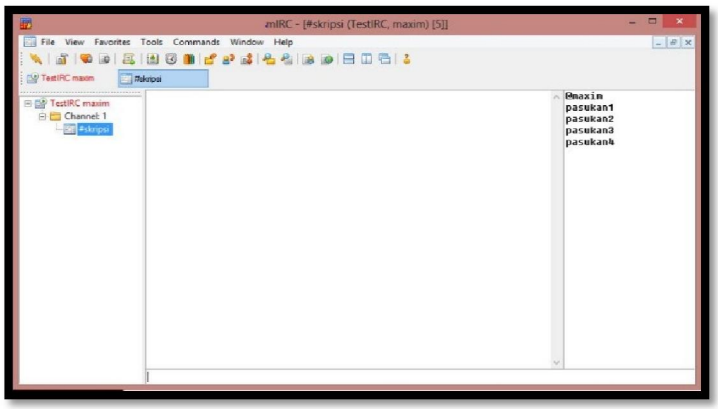


Gambar 4.29 menjalankan script zombie HTTP flood di web browser os virtual

Berdasarkan gambar 4.29 bahwa setiap script zombie dijalankan melalui web browser pada setiap sistem operasi di virtual. Untuk menjalankan script zombie pada web browser dengan cara mengetikkan perintah sebagai berikut :

1. Komputer 1 = 192.168.1.3/prod.pl = IP tersebut merupakan ip dari komputer 1 di virtual sedangkan /prod.pl merupakan file dari script zombie HTTP Flood.

2. Komputer 2 = 192.168.1.4/prod.pl = IP tersebut merupakan ip dari komputer 2 di virtual sedangkan /prod.pl merupakan file dari script zombie HTTP Flood.
3. Komputer 3 = 192.168.1.5/prod.pl = IP tersebut merupakan ip dari komputer 3 di virtual sedangkan /prod.pl merupakan file dari script zombie HTTP Flood.
4. Komputer 4 = 192.168.1.5/prod.pl = IP tersebut merupakan ip dari komputer 4 di virtual sedangkan /prod.pl merupakan file dari script zombie HTTP Flood.



Gambar 4.30 Tampilan botnet HTTP Flood di Mirc

Berdasarkan gambar 4.30 bahwa setelah melakukan load script zombie pada web browser mulai dari komputer 1 sampai komputer 4 di virtual vmware, maka pada Mirc akan tampil 4 botnet (robot & network). Berikut merupakan penjelasannya :

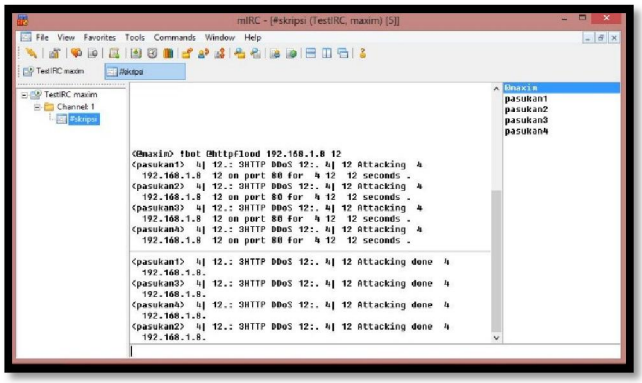
a. @maxim = merupakan nama administrator dari channel mirc yang memiliki banyak kebebasan untuk mengatur channel serta menjalankan atau memberi perintah dan mengeluarkan botnet.
b. pasukan1 = merupakan nama botnet dari komputer 1 di virtual vmware yang dijalankan melalui web browser dengan mengetik 192.168.1.3/prod.pl pada

address bar. Nama botnet tersebut bisa diganti sesuai dengan keinginan pada file prod.pl yang berada pada folder htdocs.

c. pasukan2 = merupakan nama botnet dari komputer 2 di virtual vmware yang dijalankan melalui web browser dengan mengetik 192.168.1.4/prod.pl pada address bar. Nama botnet tersebut bisa diganti sesuai dengan keinginan pada file prod.pl yang berada pada folder htdocs.

d. pasukan3 = merupakan nama botnet dari komputer 3 di virtual vmware yang dijalankan melalui web browser dengan mengetik 192.168.1.5/prod.pl pada address bar. Nama botnet tersebut bisa diganti sesuai dengan keinginan pada file prod.pl yang berada pada folder htdocs.

e. pasukan4 = merupakan nama botnet dari komputer 4 di virtual vmware yang dijalankan melalui web browser dengan mengetik 192.168.1.6/prod.pl pada address bar. Nama botnet tersebut bisa diganti sesuai dengan keinginan pada file prod.pl yang berada pada folder htdocs.



Gambar 4.31 Penyerangan HTTP Flood ke web server korban

Berdasarkan gambar 4.31 penyerangan HTTP Flood dilakukan dengan perintah :

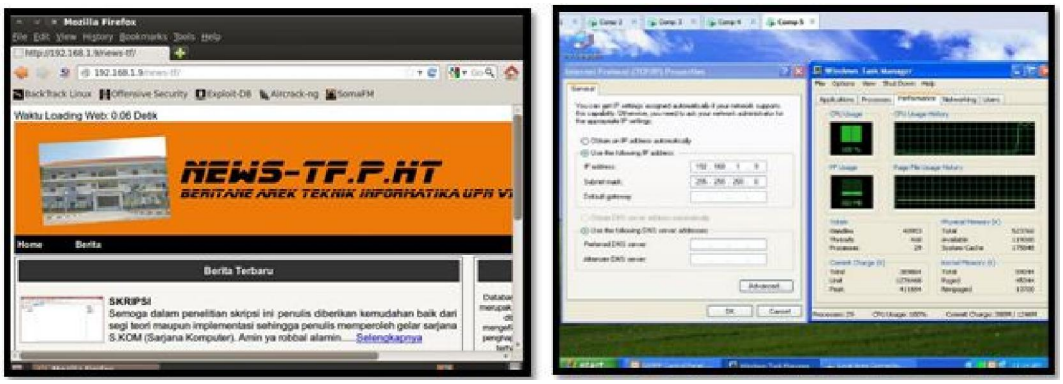
!bot @httpflood 192.168.1.8 12

- !bot merupakan perintah untuk memanggil botnet yang berada pada Mirc.

- @httpflood merupakan jenis serangan yang ada pada script zombie tersebut.
- 192.168.1.9 merupakan ip target yang akan dilakukan penyerangan.
- 12 merupakan time/waktu yang akan digunakan untuk lama penyerangan, dengan format detik, jadi setiap waktu yang diberikan akan dijalankan selama beberapa detik.

Gambar 4.32 Kondisi web server dan cpu usage sebelum diserang HTTP flood

Berdasarkan Gambar 4.32 merupakan kondisi web server dan cpu usage pada komputer 5 di sistem operasi windows xp pada virtual vmware sebelum serangan HTTP Flood dijalankan. sebelum diserang HTTP Flood saat user lain melakukan akses ke web begitu cepat hal ini dapat dibuktikan dengan adanya waktu loading web yang berada pada sebelah kiri atas web. Waktu loading web memerlukan waktu 0,01 detik (1 milisecond) dan kondisi cpu usage pada sistem operasi windows xp di virtual vmware sebelum diserang HTTP Flood tidak terjadi peningkatan.



Gambar 4.33 kondisi web server dan cpu usage saat diserang HTTP Flood

Berdasarkan gambar 4.33 merupakan kondisi web server dan cpu usage pada komputer 5 di sistem operasi windows xp pada virtual vmware saat serangan DDoS HTTP Flood dijalankan. Saat dilakukan serangan HTTP Flood dengan empat botnet pada web server tersebut saat user lain melakukan akses ke web juga tidak mengalami perubahan waktu akses yang lama. Hal ini bisa dibuktikan dengan adanya waktu loading web yang berada pada sebelah kiri atas web. Waktu loading web memerlukan waktu 0,06 detik (6 milisecond). akan tetapi kondisi cpu usage tersebut mengalami peningkatan. hampir dipastikan mengalami peningkatan yang sangat besar dari kondisi normal (sebelum diserang). Sesuai dengan parameter keberhasilan yang sudah ditetapkan maka serangan DDoS HTTP Flood dianggap sudah dijalankan dengan benar.

4.2.2 Skenario Serangan Kedua Ke Web server Komputer 5

Skenario serangan pertama dilakukan penyerangan ke web server dengan jenis serangan SYN Flood. hal ini untuk membuktikan bahwa serangan DDoS tersebut sudah dijalankan dengan benar ke web server korban, juga untuk membuktikan sebelum serangan dan saat serangan dijalankan, web server pada komputer 5 sistem operasi windows xp di virtual vmware tersebut akan meresponnya.

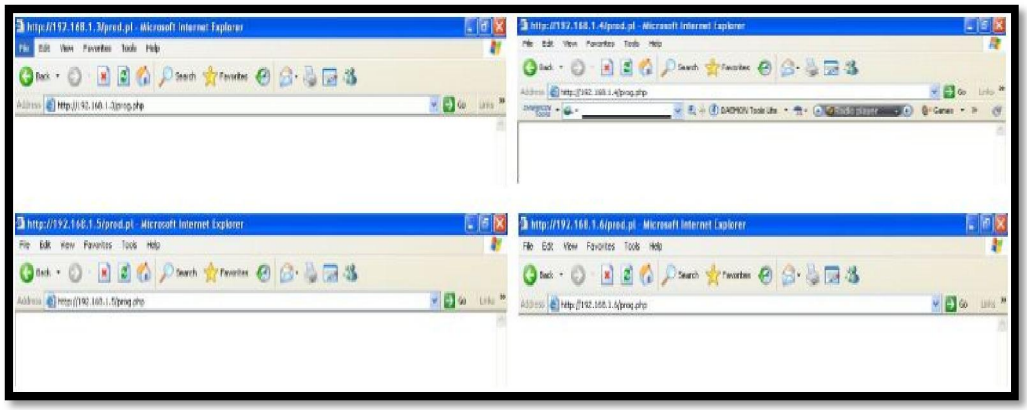
1. Serangan DDoS SYN Flood

Nama Skenario : Serangan DDoS SYN Flood pada web server
Korban pada komputer 5.

Tujuan : Skenario serangan ini bertujuan untuk
Membuktikan bahwa serangan SYN Flood
telah dijalankan dengan benar

Kebutuhan Awal : Web server pada sistem operasi windows xp di
virtual Vmware harus bisa diakses. Script zombie
harus dijalankan pada masing –masing web browser, 4 botnet harus di load pada mirc

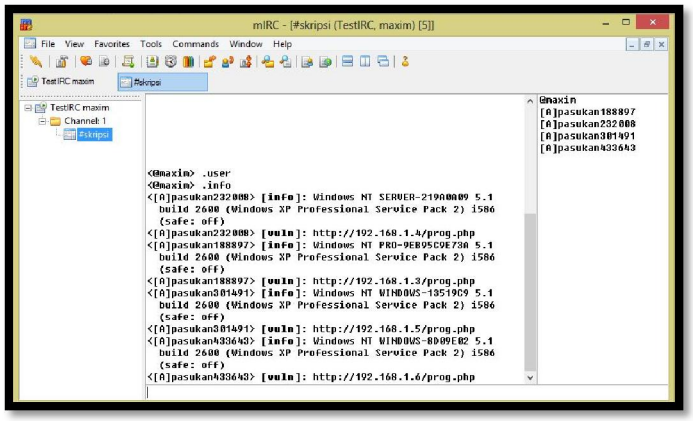
Parameter Keberhasilan : Dikatakan berhasil saat dilakukan serangan
SYN Flood ketika user lain melakukan akses ke
web server tersebut mengalami akses yang
lambat lebih dari 1,2 detik (1 detik 2 milisecond).



Gambar 4.34 menjalankan script zombie SYN Flood di web browser os virtual

Berdasarkan gambar 4.34 bahwa setiap script zombie dijalankan melalui web browser pada setiap sistem operasi di virtual. Untuk menjalankan script zombie pada web browser dengan cara mengetikkan perintah sebagai berikut :

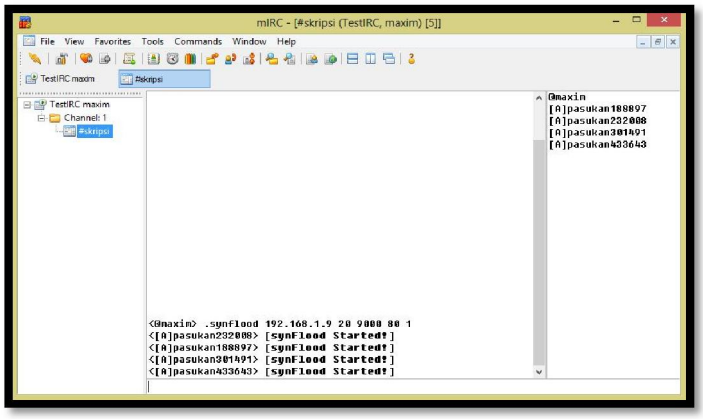
1. Komputer 1 = 192.168.1.3/prog.php = IP tersebut merupakan ip dari komputer 1 di virtual sedangkan /prog.php merupakan file dari script zombie SYN Flood.
2. Komputer 2 = 192.168.1.4/prog.php = IP tersebut merupakan ip dari komputer 2 di virtual sedangkan /prog.php merupakan file dari script zombie SYN Flood.
3. Komputer 3 = 192.168.1.5/prog.php = IP tersebut merupakan ip dari komputer 3 di virtual sedangkan /prog.php merupakan file dari script zombie SYN Flood.
4. Komputer 4 = 192.168.1.5/prog.php = IP tersebut merupakan ip dari komputer 4 di virtual sedangkan /prog.php merupakan file dari script zombie SYN Flood.



Gambar 4.35 Tampilan botnet SYN Flood di Mirc

Berdasarkan gambar 4.35 bahwa setelah melakukan load script zombie pada web browser mulai dari komputer 1 sampai komputer 4 di virtual vmware, maka pada mirc akan tampil 4 botnet (robot & network). Berikut merupakan penjelasannya :

a. @maxim merupakan nama administrator dari channel mirc yang memiliki banyak kebebasan untuk mengatur channel serta menjalankan atau memberi perintah dan mengeluarkan botnet.
b. [A] pasukan 188897 = merupakan nama botnet dari komputer 1 di virtual vmware yang dijalankan melalui web browser dengan mengetik 192.168.1.3/prog.php pada address bar.
c. [A] pasukan 232008 merupakan nama botnet dari komputer 2 di virtual vmware yang dijalankan melalui web browser dengan mengetik 192.168.1.4/prog.php pada address bar.
d. [A] pasukan 301491 merupakan nama botnet dari komputer 3 di virtual vmware yang dijalankan melalui web browser dengan mengetik 192.168.1.5/prog.php pada address bar.
e. [A] pasukan 433643 merupakan nama botnet dari komputer 4 di virtual vmware yang dijalankan melalui web browser dengan mengetik 192.168.1.5/prog.php pada address bar.

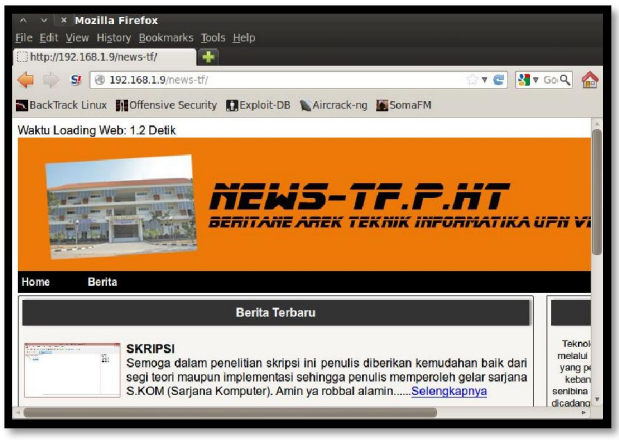


Gambar 4.36 Penyerangan SYN Flood ke web server korban

Berdasarkan gambar 4.36 penyerangan HTTP Flood dilakukan dengan perintah :

.synflood 192.168.1.9 20 9000 1

- .synflood merupakan jenis serangan yang ada pada script zombie tersebut.
- 192.168.1.9 merupakan ip target yang akan dilakukan penyerangan.
- 20 merupakan jumlah paket yang akan dikirim ke web server korban.
- 9000 merupakan besar paket yang akan dikirim ke web server korban.
- 80 merupakan port yang akan diserang oleh botnet.
- 1 merupakan delay atau waktu tunggu setiap paket yang akan terkirim.



Gambar 4.37 kondisi web server sebelum diserang SYN Flood

Berdasarkan Gambar 4.37 merupakan kondisi web server di virtual vmware sebelum dilakukan penyerangan. sebelum diserang SYN Flood saat user lain

melakukan akses ke web begitu cepat hal ini dapat dibuktikan dengan adanya waktu loading web yang berada pada sebelah kiri atas web. Waktu loading web memerlukan waktu 1.2 detik (1 detik 2 milisecond).

Gambar 4.38 Kondisi web server saat diserang SYN Flood

Berdasarkan gambar 4.38 merupakan kondisi web server dan cpu usage di sistem operasi windows xp pada virtual vmware saat serangan SYN Flood dijalankan. Saat dilakukan serangan SYN Flood dengan empat botnet pada web server tersebut saat user lain melakukan akses ke web tersebut agak sedikit lama. Hal ini bisa dibuktikan dengan adanya waktu loading web yang berada pada sebelah kiri atas web. Waktu loading web memerlukan waktu 10,95 detik (10 detik 95 milisecond)

Sesuai dengan parameter keberhasilan yang sudah ditetapkan maka serangan DDoS SYN Flood dianggap sudah dijalankan dengan benar.

4.3 IMPLEMENTASI SKENARIO SERANGAN 2

Skenario serangan 2 merupakan skenario inti dari serangan DDoS HTTP Flood dan SYN Flood, bertujuan untuk membuktikan bahwa serangan DDoS HTTP Flood dan SYN Flood sudah dijalankan ke Honeyd, nantinya file log Honeyd akan mencatat serangan-serangannya dan membuktikan traffic normal dan traffic serangan.

4.3.1 Skenario Serangan pertama Ke Honeyd

Skenario serangan pertama dilakukan penyerangan ke Honeyd dengan jenis serangan HTTP Flood. hal ini untuk membuktikan bahwa serangan yang dilakukan ke Honeyd berjalan dengan benar. Langkah pertama untuk menjalankan skenario serangan ini adalah dengan cara menjalankan script zombie pada masing-masing web browser dari setiap sistem operasi windows xp di virtual vmware, nantinya pada Mirc (command and control) server akan tampil 4 botnet (robot and network).

1. Serangan DDoS HTTP Flood

Nama Skenario : Serangan DDoS HTTP Flood pada Honeyd

Tujuan : Skenario serangan ini bertujuan untuk

Membuktikan bahwa serangan HTTP Flood

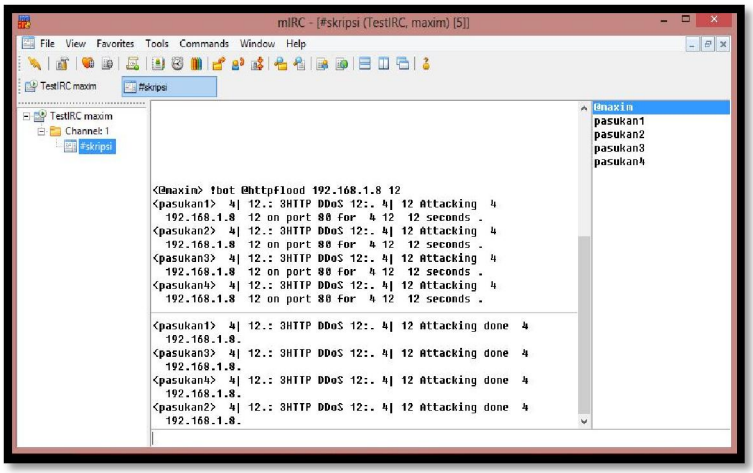
Berjalan dengan benar dan bisa terdeteksi oleh Honeyd.

Kebutuhan Awal : Script zombie harus dijalankan pada web browser,

4 botnet (robot dan network) harus di load pada Mirc

Parameter Keberhasilan : Dikatakan berhasil saat Honeyd bisa mendeteksi

Alamat IP Attacker (penyerang), port 80 sebagai tujuan penyerangan Serta paket yang diterima oleh Honeyd jumlahnya sangat banyak dan besar.



Gambar 4.39 penyerangan HTTP Flood ke Honeyd

Berdasarkan gambar 4.39 adalah tampilan botnet pada Mirc, ke-empat botnet tersebut akan diperintah untuk menyerang Honeyd pada port 80.

```
2013-12-22-16:24:55.6613 tcp(6) E 192.168.1.4 3469 192.168.1.8 80: 74
2013-12-22-16:24:55.6620 tcp(6) E 192.168.1.6 3513 192.168.1.8 80: 74
2013-12-22-16:24:55.6622 tcp(6) E 192.168.1.5 3490 192.168.1.8 80: 74
2013-12-22-16:24:55.6633 tcp(6) S 192.168.1.4 3471 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:55.6646 tcp(6) S 192.168.1.6 3515 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:55.6650 tcp(6) S 192.168.1.5 3492 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:55.6655 tcp(6) E 192.168.1.3 2307 192.168.1.8 80: 74
2013-12-22-16:24:55.6659 tcp(6) E 192.168.1.4 3470 192.168.1.8 80: 74
2013-12-22-16:24:55.6662 tcp(6) S 192.168.1.3 2308 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:55.6665 tcp(6) E 192.168.1.6 3514 192.168.1.8 80: 74
2013-12-22-16:24:55.6674 tcp(6) E 192.168.1.5 3491 192.168.1.8 80: 74
2013-12-22-16:24:55.6683 tcp(6) S 192.168.1.4 3472 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:55.6712 tcp(6) S 192.168.1.6 3516 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:55.6715 tcp(6) S 192.168.1.5 3493 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:55.6737 tcp(6) E 192.168.1.4 3471 192.168.1.8 80: 74
2013-12-22-16:24:55.6740 tcp(6) S 192.168.1.3 2309 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:55.6749 tcp(6) E 192.168.1.6 3515 192.168.1.8 80: 74
2013-12-22-16:24:55.6752 tcp(6) S 192.168.1.4 3473 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:55.6755 tcp(6) E 192.168.1.5 3492 192.168.1.8 80: 74
```

Gambar 4.40 Log Honeyd terhadap serangan HTTP Flood

Gambar 4.40 merupakan salah satu contoh potongan log Honeyd terhadap serangan HTTP Flood, berikut penjelasan dari log Honeyd tersebut :

1. Log dijalankan pada tanggal 22-12-2013.
2. Perbedaan waktu yang singkat yang dilakukan oleh Attacker untuk melakukan penyerangan. hal ini bisa ditunjukkan pada waktu terjadinya event tersebut pada jam 16:24:55.
3. Protokol yang digunakan adalah TCP
4. S dan E merupakan kepanjangan dari start new connection (membuat sebuah koneksi baru) dan end of a connection (mengakhiri sebuah koneksi) nantinya paket yang diterima honeyd akan terjadi pada simbol E ini.
5. Alamat IP 192.168.1.3, 192.168.1.4, 192.168.1.5, 192.168.1.6 merupakan alamat IP yang digunakan Attacker untuk menyerang Honeyd.
6. Alamat IP yang diserang oleh Attacker adalah alamat IP Honeyd 192.168.1.8 pada port 80.
7. Besar paket yang diterima oleh Honeyd setiap end of connection sebesar 74 Byte (besar paket ini akan terus bertambah berkali lipat, tergantung pada file log Honeyd).

Sesuai dengan parameter keberhasilan yang sudah ditetapkan maka pada penyerangan HTTP Flood ke Honeyd dikatakan berhasil.

4.3.2 Skenario Serangan kedua Ke Honeyd

Skenario serangan kedua dilakukan penyerangan ke Honeyd dengan jenis serangan SYN Flood. hal ini untuk membuktikan bahwa serangan yang dilakukan ke Honeyd berjalan dengan benar. Langkah pertama untuk menjalankan skenario serangan ini adalah dengan cara menjalankan script zombie pada web browser

dari setiap sistem operasi windows xp di virtual vmware, nantinya pada Mirc (command and control) server akan tampil 4 botnet (robot and network).

1. Serangan DDoS SYN Flood

Nama Skenario : Serangan DDoS SYN Flood pada Honeyd

Tujuan : Skenario serangan ini bertujuan untuk

Membuktikan bahwa serangan SYN Flood Berjalan dengan benar dan bisa terdeteksi oleh Honeyd.

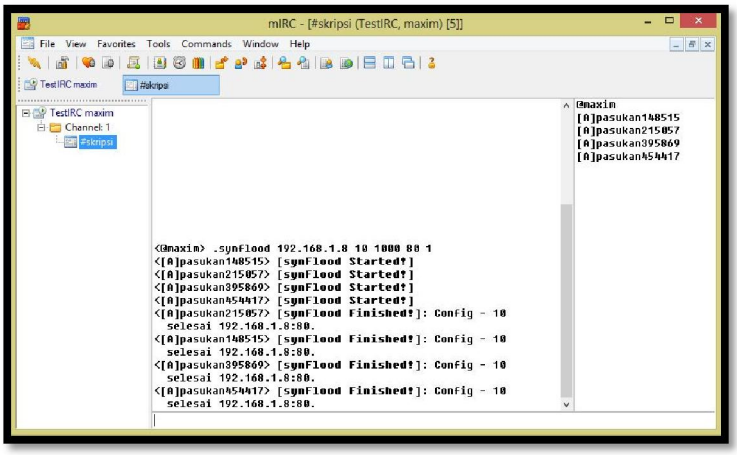
Kebutuhan Awal : Script zombie harus dijalankan pada web browser,

4 botnet harus di load pada Mirc.

Parameter Keberhasilan : Dikatakan berhasil saat Honeyd bisa mendeteksi

Alamat IP Attacker (penyerang) dan port 80 sebagai tujuan penyerangan, serta paket yang yang diterima oleh Honeyd jumlahnya sangat banyak dan besar.

Berikut adalah langkah-langkah untuk melakukan penyerangan SYN flood ke Honeyd :



Gambar 4.41 Penyerangan SYN Flood ke Honeyd

Berdasarkan gambar 4.41 adalah tampilan botnet pada Mirc, ke-empat botnet tersebut akan diperintah untuk menyerang Honeyd pada port 80.

```
2013-12-22-17:54:16.5946 tcp(6) S 192.168.1.3 1080 192.168.1.8 80
[Windows XP SP1]
2013-12-22-17:54:16.5956 tcp(6) S 192.168.1.6 1076 192.168.1.8 80
[Windows XP SP1]
2013-12-22-17:54:16.6039 tcp(6) S 192.168.1.5 1076 192.168.1.8 80
[Windows XP SP1]
2013-12-22-17:54:16.6865 tcp(6) E 192.168.1.3 1080 192.168.1.8 80: 1000
4096
2013-12-22-17:54:16.6867 tcp(6) - 192.168.1.3 1080 192.168.1.8 80: 40 R
[Windows XP SP1]
2013-12-22-17:54:16.6867 tcp(6) - 192.168.1.3 1080 192.168.1.8 80: 40 R
[Windows XP SP1]
2013-12-22-17:54:16.6867 tcp(6) - 192.168.1.3 1080 192.168.1.8 80: 40 R
[Windows XP SP1]
2013-12-22-17:54:16.6867 tcp(6) - 192.168.1.3 1080 192.168.1.8 80: 40 R
[Windows XP SP1]
2013-12-22-17:54:16.6867 tcp(6) - 192.168.1.3 1080 192.168.1.8 80: 40 R
[Windows XP SP1]
2013-12-22-17:54:16.6868 tcp(6) - 192.168.1.3 1080 192.168.1.8 80: 40 R
[Windows XP SP1]
2013-12-22-17:54:16.6868 tcp(6) - 192.168.1.3 1080 192.168.1.8 80: 40 R
[Windows XP SP1]
2013-12-22-17:54:16.7059 tcp(6) E 192.168.1.5 1076 192.168.1.8 80: 1000
2048
2013-12-22-17:54:16.7065 tcp(6) - 192.168.1.5 1076 192.168.1.8 80: 40 R
[Windows XP SP1]
```

Gambar 4.42 Log Honeyd terhadap serangan SYN Flood

Gambar 4.42 merupakan salah satu contoh potongan log Honeyd terhadap serangan SYN Flood, berikut penjelasan dari log Honeyd tersebut :

1. Log dijalankan pada tanggal 22-12-2013
2. Perbedaan waktu yang singkat yang dilakukan oleh Attacker untuk melakukan penyerangan. hal ini bisa ditunjukkan pada waktu terjadinya event tersebut pada jam 17:54:16.

3. Protokol yang digunakan adalah TCP.
4. S dan E merupakan kepanjangan dari start new connection (membuat sebuah koneksi baru) dan end of a connection (mengakhiri sebuah koneksi) nantinya paket yang diterima honeyd akan terjadi pada simbol E ini.
5. Alamat IP 192.168.1.3, 192.168.1.4, 192.168.1.5, 192.168.1.6 merupakan alamat IP yang digunakan Attacker untuk menyerang Honeyd.
6. Alamat IP yang diserang adalah alamat IP Honeyd 192.168.1.8 pada port 80.
7. Besar paket yang diterima oleh Honeyd setiap end of a connection sebesar 1000 Byte (besar paket ini akan terus bertambah berkali lipat, tergantung pada file log Honeyd)

Sesuai dengan parameter keberhasilan yang sudah ditetapkan maka pada penyerangan SYN Flood ke Honeyd dikatakan berhasil.

4.4 PENDETEKSIAN DAN ANALISA SERANGAN

Pendeteksian dan analisa serangan dilakukan setelah semua skenario serangan dilakukan agar untuk membuktikan bahwa serangan DDoS HTTP Flood dan SYN Flood dilakukan dengan benar nantinya akan ada 20 kali percobaan untuk setiap jenis serangan. Total percobaan dari jenis serangan tersebut sebanyak 40 kali percobaan.

4.4.1 Pendeteksian Dan Analisa Serangan DDoS HTTP Flood

Nama Skenario : Serangan DDoS HTTP Flood pada Honeyd

Tujuan : Skenario serangan ini bertujuan untuk

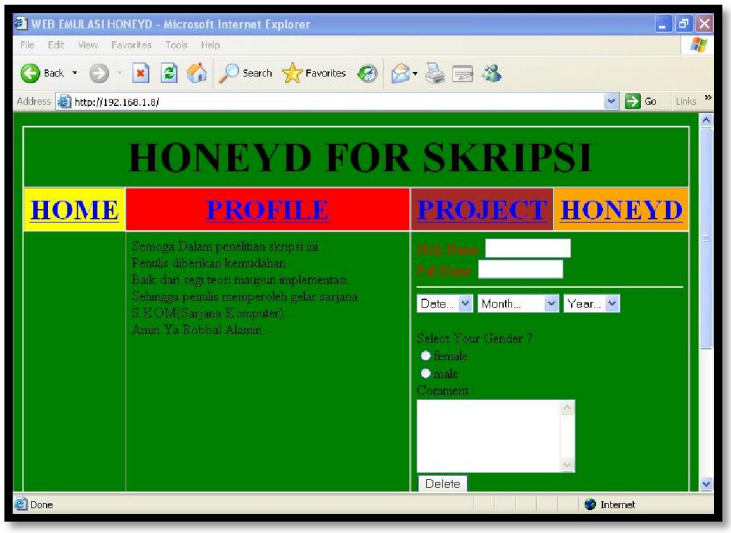
Membuktikan perbedaan traffic normal dan traffic serangan

Kebutuhan Awal : Script zombie harus dijalankan pada web browser, 4 botnet harus di load pada Mirc.

Parameter Keberhasilan: Dikatakan berhasil saat alamat IP user yang melakukan akses normal dan alamat IP Attacker (Penyerang) yang melakukan akses serangan memenuhi beberapa kriteria.

1. Jumlah paket yang diterima oleh Honeyd ketika user lain melakukan akses normal sedikit sedangkan Attacker yang melakukan serangan DDoS HTTP Flood sangat banyak.
2. Besar paket yang diterima oleh Honeyd saat user lain melakukan akses normal sedikit sedangkan besar paket yang diterima oleh Honeyd saat Attacker melakukan serangan DDoS HTTP Flood sangat besar.
3. Terjadinya perbedaan waktu sekitar 1-2 detik yang dilakukan user untuk melakukan akses normal ke web emulasi Honeyd sedangkan Attacker saat menyerang web emulasi Honeyd tersebut memerlukan waktu yang sangat singkat.

Untuk memudahkan dalam implementasi maka Langkah awal akan dilakukan akses normal ke web yang di emulasi oleh Honeyd untuk membuktikan traffic normal yang telah dilakukan.



Gambar 4.43 Tampilan web yang diemulasi Honeyd

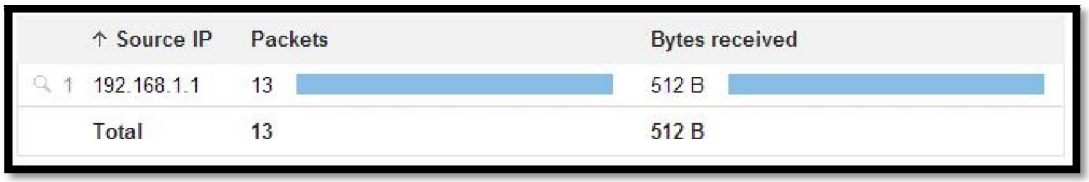
Saat user melakukan akses ke web emulasi Honeyd maka hasilnya terlihat seperti gambar web pada gambar 4.43, maka secara otomatis Honeyd akan mendeteksinya secara real time dan file log pada Honeyd akan mencatatnya. Tampilan file log Honeyd akan dijelaskan pada gambar 4.44.

```
2013-12-22-16:34:52.3844 tcp(6) S 192.168.1.1 51336 192.168.1.8 80
2013-12-22-16:34:53.5793 tcp(6) S 192.168.1.1 51334 192.168.1.8 80
2013-12-22-16:34:53.5913 tcp(6) S 192.168.1.1 51335 192.168.1.8 80
2013-12-22-16:34:54.8786 tcp(6) S 192.168.1.1 51337 192.168.1.8 80
2013-12-22-16:34:55.8139 tcp(6) S 192.168.1.1 51338 192.168.1.8 80
2013-12-22-16:34:56.9848 tcp(6) S 192.168.1.1 51339 192.168.1.8 80
2013-12-22-16:34:58.2970 tcp(6) S 192.168.1.1 51340 192.168.1.8 80
2013-12-22-16:34:59.1921 tcp(6) S 192.168.1.1 51341 192.168.1.8 80
2013-12-22-16:35:01.5246 tcp(6) S 192.168.1.1 51342 192.168.1.8 80
2013-12-22-16:35:03.5230 tcp(6) S 192.168.1.1 51343 192.168.1.8 80
2013-12-22-16:35:05.7285 tcp(6) S 192.168.1.1 51344 192.168.1.8 80
2013-12-22-16:35:08.7230 tcp(6) S 192.168.1.1 51345 192.168.1.8 80
2013-12-22-16:35:18.7355 tcp(6) E 192.168.1.1 51345 192.168.1.8 80: 0 512
```

Gambar 4.44 Tampilan file log web emulasi Honeyd terhadap akses normal

Berdasarkan gambar 4.44 merupakan tampilan file log Honeyd terhadap akses normal. Terlihat hanya Alamat IP 192.168.1.1 yang melakukan akses ke web emulasi Honeyd, juga terdapat perbedaan waktu 1 sampai 2 detik saat terjadinya event tersebut. Hal ini membuktikan bahwa user tersebut hanya melakukan akses normal, karena ada perbedaan waktu tersebut. Untuk memahami file log seperti

gambar 4.44 cukup sulit, Oleh karena itu dari file log tersebut akan diolah menjadi tampilan grafis seperti dibawah ini :

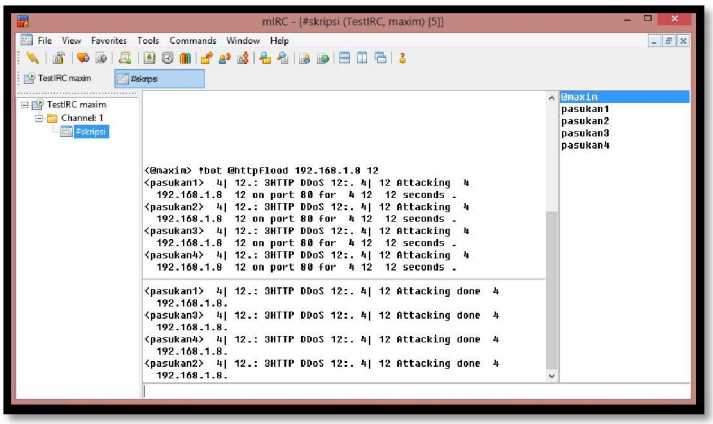


	↑ Source IP	Packets	Bytes received
1	192.168.1.1	13	512 B
	Total	13	512 B

Gambar 4.45 Tampilan grafis setelah user melakukan akses normal

Berdasarkan gambar 4.45 pada tampilan grafis tersebut dapat menjelaskan bahwa saat user melakukan akses normal, jumlah paket yang diterima oleh Honeyd sebanyak 13 paket dan besar paket yang diterima oleh Honeyd sebesar 512 Byte.

Setelah pendeteksian, dan mengolah file log pada Honeyd maka sekarang tibalah saatnya untuk melancarkan serangan DDoS HTTP Flood.



Gambar 4.46 menjalankan serangan HTTP Flood untuk analisa

Pada gambar 4.46 merupakan tampilan botnet pada Mirc, ke-empat botnet tersebut akan diperintah untuk menyerang IP 192.168.1.8 pada Honeyd. Penyerangan tersebut akan berlangsung selama 12 detik.

```
2013-12-22-16:24:59.9390 tcp(6) E 192.168.1.3 2660 192.168.1.8 80: 74 0
2013-12-22-16:24:59.9404 tcp(6) E 192.168.1.4 4129 192.168.1.8 80: 74 0
2013-12-22-16:24:59.9408 tcp(6) S 192.168.1.6 4177 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:59.9410 tcp(6) E 192.168.1.5 4149 192.168.1.8 80: 74 0
2013-12-22-16:24:59.9425 tcp(6) S 192.168.1.4 4131 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:59.9430 tcp(6) S 192.168.1.5 4151 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:59.9433 tcp(6) E 192.168.1.6 4176 192.168.1.8 80: 74 0
2013-12-22-16:24:59.9435 tcp(6) S 192.168.1.3 2662 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:59.9444 tcp(6) E 192.168.1.4 4130 192.168.1.8 80: 74 0
2013-12-22-16:24:59.9447 tcp(6) S 192.168.1.6 4178 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:59.9449 tcp(6) E 192.168.1.5 4150 192.168.1.8 80: 74 0
2013-12-22-16:24:59.9452 tcp(6) E 192.168.1.3 2661 192.168.1.8 80: 74 0
2013-12-22-16:24:59.9462 tcp(6) S 192.168.1.4 4132 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:59.9465 tcp(6) E 192.168.1.6 4177 192.168.1.8 80: 74 0
2013-12-22-16:24:59.9466 tcp(6) S 192.168.1.5 4152 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:59.9560 tcp(6) E 192.168.1.4 4131 192.168.1.8 80: 74 0
2013-12-22-16:24:59.9563 tcp(6) E 192.168.1.5 4151 192.168.1.8 80: 74 0
2013-12-22-16:24:59.9565 tcp(6) S 192.168.1.6 4179 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:59.9584 tcp(6) S 192.168.1.4 4133 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:59.9588 tcp(6) S 192.168.1.5 4153 192.168.1.8 80
[Windows XP SP1]
2013-12-22-16:24:59.9592 tcp(6) E 192.168.1.6 4178 192.168.1.8 80: 74 0
```

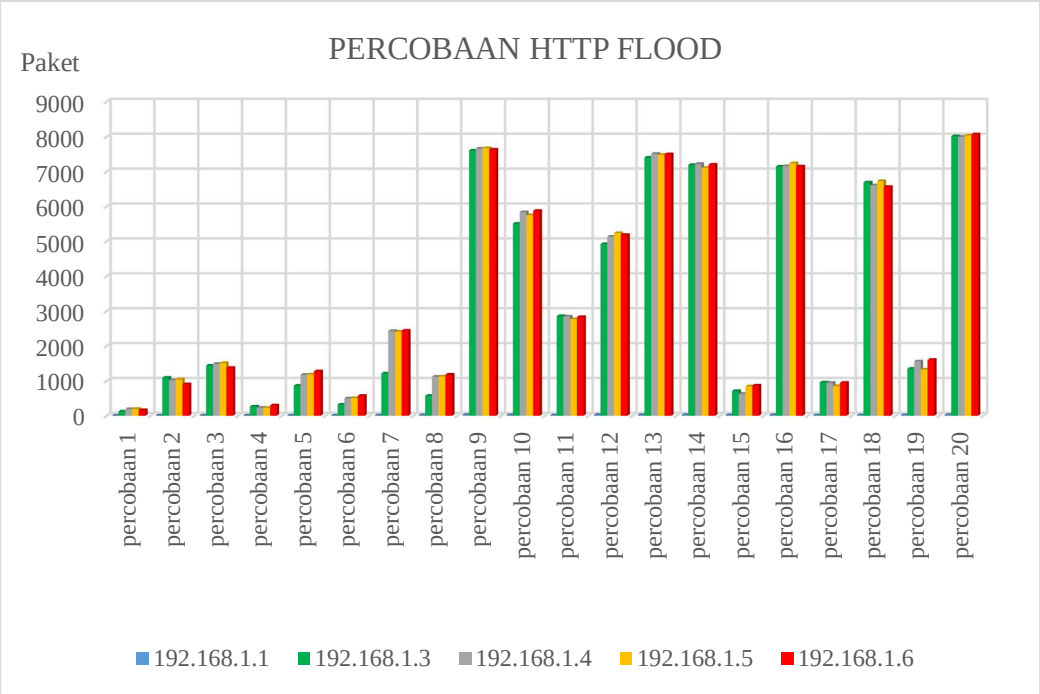
Gambar 4.47 Tampilan file log Honeyd terhadap serangan DDoS HTTP Flood

Pada gambar 4.47 merupakan potongan log pada Honeyd ketika serangan HTTP Flood dijalankan. untuk memahami log tersebut akan diolah menjadi tampilan grafis seperti gambar 4.48.

	↑ Source IP	Packets	Bytes received
1	192.168.1.1	12	20.13 K
2	192.168.1.3	1,214	45.50 K
3	192.168.1.4	2,432	93.00 K
4	192.168.1.5	2,414	82.50 K
5	192.168.1.6	2,441	89.00 K
	Total	8,513	330.13 K

Gambar 4.48 Tampilan grafis sesudah serangan DDoS HTTP Flood

Pada gambar 4.48 terjadi perbedaan yang sangat jauh ketika perbedaan paket yang diterima oleh Honeyd pada setiap alamat IP yang mengakses web emulasi Honeyd. untuk lebih membuktikan hasil analisa serangan DDoS HTTP Flood akan dilakukan percobaaan sebanyak 20 kali percobaan. Berikut adalah tampilan diagram untuk setiap 20 kali percobaan serangan DDoS HTTP Flood :



Gambar 4.49 Jumlah paket serangan DDoS HTTP Flood percobaan 1 sampai 20

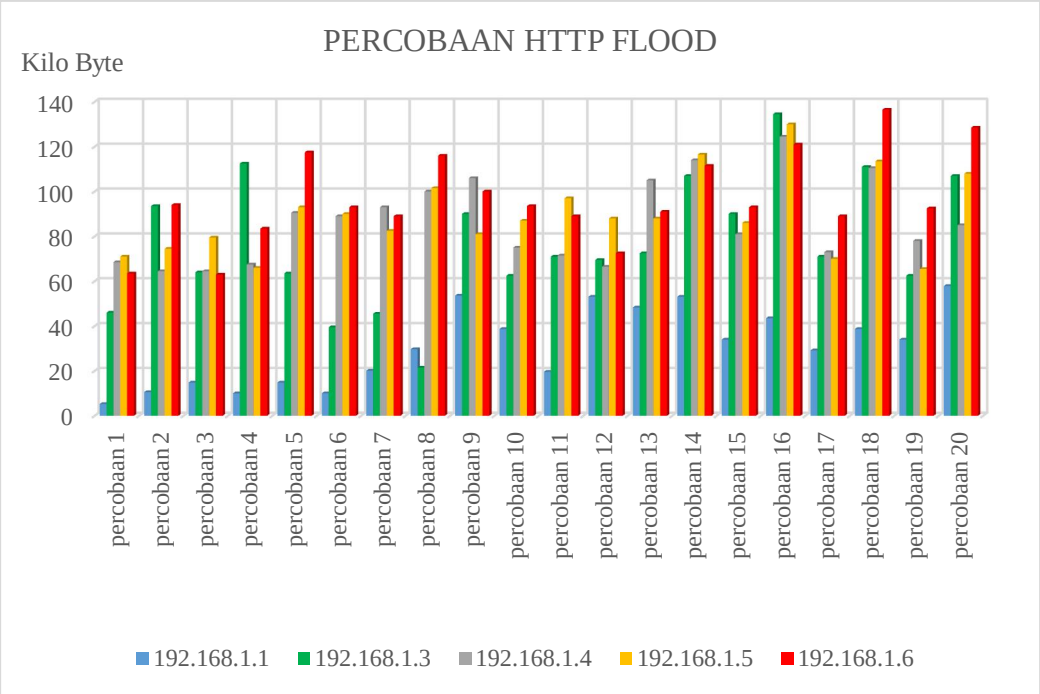
Untuk memudahkan dalam membaca jumlah paket yang diterima oleh Honeyd maka akan ditabelkan seperti pada tabel 4.1.

Tabel 4.1 Jumlah paket serangan DDoS HTTP Flood percobaan 1 sampai 20

PERCOBAAN HTTP FLOOD	ALAMAT IP				
	192.168.1.1	192.168.1.3	192.168.1.4	192.168.1.5	192.168.1.6
1	4	124	189	197	164
2	8	1093	1021	1043	903
3	8	1439	1489	1509	1371
4	6	267	234	230	298
5	8	862	1174	1186	1276
6	6	322	499	509	571
7	12	1214	2432	2414	2441

8	17	573	1118	1126	1181
9	27	7605	7657	7673	7630
10	18	5507	5839	5748	5874
11	10	2856	2846	2772	2832
12	24	4921	5136	5237	5185
13	22	4921	5136	5237	5185
14	24	4921	5136	5237	5185
15	16	709	634	842	869
16	20	7144	7158	7238	7147
17	14	956	944	854	944
18	18	6690	6606	6727	6558
19	16	1348	1559	1328	1601
20	26	8012	7997	8035	8067

Dari gambar dan tabel tersebut saat User melakukan akses normal dengan alamat IP 192.168.1.1 pada Honeyd menerima puluhan paket dan saat Attacker dengan Alamat IP 192.168.1.3 sampai 192.168.1.6 pada Honeyd menerima ratusan hingga ribuan oleh setiap alamat IP ke web emulasi Honeyd. setelah dilakukan 20 kali percobaan serangan DDoS HTTP Flood pada Honeyd, sekarang saatnya untuk mengetahui besar paket yang diterima oleh Honeyd, berikut adalah gambar diagramnya :



Gambar 4.50 Besar paket serangan DDoS HTTP Flood percobaan 1 sampai 20

Untuk memudahkan dalam membaca besar paket yang diterima oleh Honeyd, maka akan ditabelkan seperti tabel dibawah ini :

Tabel 4.2 Besar paket serangan DDoS HTTP Flood percobaan 1 sampai 20

PERCOBAAN HTTP FLOOD	ALAMAT IP				
	192.168.1.1	192.168.1.3	192.168.1.4	192.168.1.5	192.168.1.6
1	5,28 KB	46 KB	68,5 KB	71 KB	63,5 KB
2	10,56 KB	93,5 KB	64,5 KB	74,5 KB	94 KB
3	14,84 KB	64 KB	64,5 KB	74,5 KB	94 KB
4	10,06 KB	112,5 KB	67,5 KB	66 KB	83,5 KB
5	14,84 KB	63,5 KB	90,5 KB	93 KB	117,5 KB
6	10,06 KB	39,5 KB	89 KB	90 KB	93 KB
7	20,13 KB	45,5 KB	93 KB	82,5 KB	89 KB
8	29,69 KB	21,5 KB	100 KB	101,5 KB	116 KB

9	53,59 KB	90 KB	106 KB	81 KB	100 KB
10	38,75 KB	62,5 KB	75 KB	87 KB	93,5 KB
11	19,63 KB	71 KB	71,5 KB	97 KB	89 KB
12	53,09 KB	69,5 KB	66,5 KB	88 KB	72,5 KB
13	48,31 KB	72,5 KB	105 KB	88 KB	91 KB
14	53,09 KB	107 KB	114 KB	116,5 KB	111,5 KB
15	33,97 KB	90 KB	81 KB	86 KB	93 KB
16	43,53 KB	134,5 KB	124,5 KB	130 KB	121 KB
17	29,19 KB	71 KB	73 KB	70 KB	89 KB
18	38,75 KB	111 KB	110,5 KB	113,5 KB	136,5 KB
19	33,97 KB	62,5 KB	78 KB	65,5 KB	92,5 KB
20	57,88 KB	107 KB	85 KB	108 KB	128,5 KB

Dari gambar dan tabel tersebut saat User melakukan akses normal dengan alamat IP 192.168.1.1, Honeyd menerima besar paket cukup sedikit dibandingkan dengan saat Attacker dengan Alamat IP 192.168.1.3 sampai 192.168.1.6 yang mengirimkan besar paket yang sangat besar ke Honeyd. Setelah semua data telah dikumpulkan saatnya untuk menyamakan dengan parameter keberhasilan yang sudah ditetapkan.

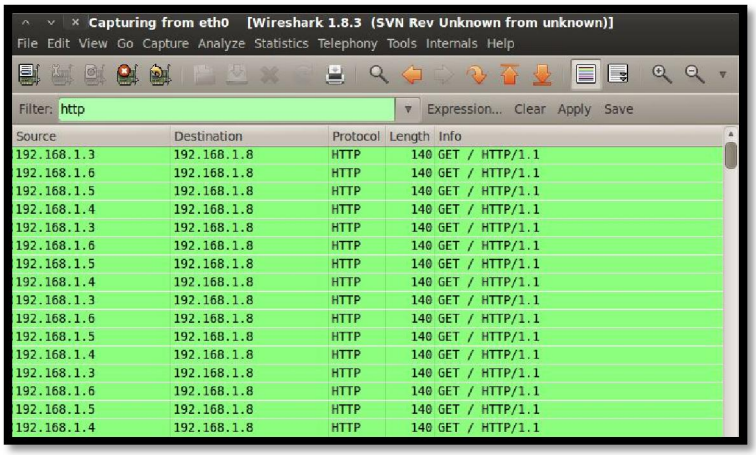
1. Jumlah paket yang diterima oleh Honeyd saat User melakukan akses normal sedikit, sedangkan jumlah paket yang diterima oleh Honeyd saat Attacker melakukan serangan DDoS sangat banyak, hal ini bisa dibuktikan dengan tabel yang sudah dibuat untuk 20 kali serangan HTTP Flood.

2. Besar paket yang diterima oleh Honeyd saat User melakukan akses normal sedikit sedangkan besar paket yang diterima oleh Honeyd saat Attacker melakukan serangan DDoS HTTP Flood sangat besar.
3. Pada file log Honeyd terhadap serangan DDoS HTTP Flood pada gambar 4.47, terdapat banyak alamat IP mulai dari 192.168.1.3, 192.168.1.4, 192.168.1.5 dan 192.168.1.6 yang melakukan akses ke web emulasi Honeyd pada waktu yang relatif sama yaitu pada pukul 16.24.59, hampir dipastikan tidak ada perbedaan waktu sedikitpun saat ke-empat alamat IP tersebut melakukan akses ke web emulasi Honeyd. Dengan demikian bisa dipastikan bahwa ke-empat alamat IP tersebut digunakan oleh Attacker untuk menyerang web emulasi Honeyd.

Untuk membuktikan persentase dari jumlah paket dan besar paket yang diterima oleh Honeyd saat akses normal dan serangan DDoS HTTP Flood, maka akan dibuat persentase berdasarkan percobaan selama 20 kali.

a. Total persentase jumlah paket dari akses normal = 0,11%
b. Total persentase jumlah paket dari serangan DDoS HTTP Flood = 99,86%
c. Total persentase besar paket dari akses normal = 8,11%
d. Total persentase besar paket dari serangan DDoS HTTP Flood = 91,87%

Dari ke-tiga ulasan dan persentase tersebut sesuai dengan parameter keberhasilan yang sudah ditetapkan, Hal ini membuktikan bahwa serangan DDoS HTTP Flood pada web emulasi Honeyd terbukti dan benar adanya. Untuk mengetahui saat proses Penyerangan DDoS HTTP Flood yang lebih spesifik ke Honeyd maka digunakan aplikasi wireshark.



Gambar 4.51 Tampilan wireshark ketika dijalankan serangan DDoS HTTP Flood

Berdasarkan Pada gambar 4.51 merupakan tampilan Wireshark ketika dijalankan serangan DDoS HTTP Flood, dimana ada keterangan berupa GET artinya bahwa mengirim sejumlah banyak permintaan yang dikirim oleh alamat IP 192.168.1.3, 192.168.1.6, 192.168.1.5, 192.168.1.4 ke web emulasi Honeyd dengan alamat IP 192.168.1.8.

4.4.2 Pendeteksian Dan Analisa Serangan DDoS SYN Flood

Nama Skenario : Serangan DDoS SYN Flood pada Honeyd

Tujuan : Skenario serangan ini bertujuan untuk membuktikan perbedaan traffic normal dan traffic serangan

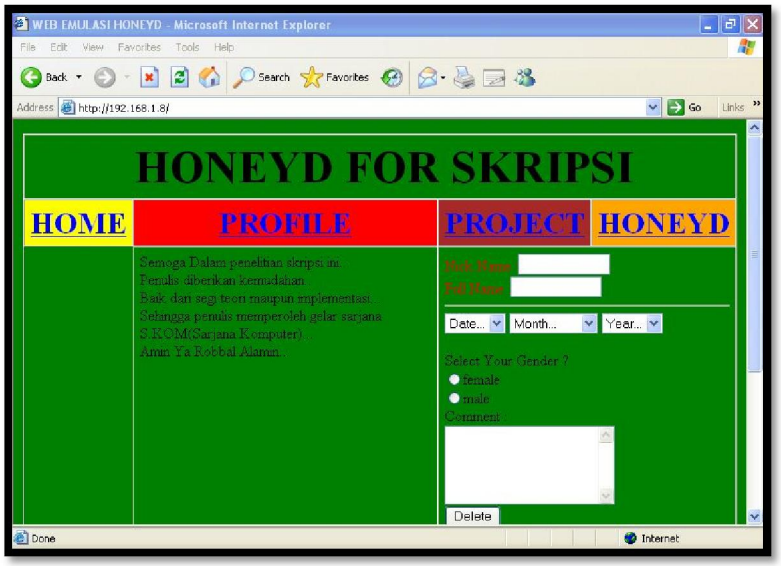
Kebutuhan Awal : Script zombie harus dijalankan pada web browser, 4 botnet harus di load pada Mirc.

Parameter Keberhasilan : Dikatakan berhasil saat alamat IP user yang melakukan akses normal dan alamat IP Attacker (Penyerang)

Yang melakukan serangan memenuhi beberapa kriteria.

1. Jumlah paket yang diterima oleh Honeyd ketika User lain melakukan akses normal sedikit sedangkan Attacker yang melakukan serangan DDoS SYN Flood sangat banyak.
2. Besar paket yang diterima oleh Honeyd saat User lain melakukan akses normal sedikit sedangkan besar paket yang diterima oleh Honeyd saat Attacker melakukan serangan DDoS SYN Flood sangat besar.
3. Terjadinya perbedaan waktu sekitar 1-2 detik yang dilakukan User untuk melakukan akses normal ke web emulasi Honeyd sedangkan Attacker saat menyerang web emulasi Honeyd tersebut memerlukan waktu yang sangat singkat.

Langkah awal akan dilakukan akses normal ke web yang di emulasi oleh Honeyd untuk membuktikan traffic normal yang telah dilakukan.



Gambar 4.52 Tampilan web yang diemulasi Honeyd

Saat User melakukan akses ke web emulasi Honeyd maka hasilnya seperti pada gambar 4.52, maka secara otomatis Honeyd akan mendeteksinya dan file log pada Honeyd akan mencatatnya, berikut adalah tampilan file log

```
2013-12-22-16:34:45.8560 honeyd log started ------
2013-12-22-16:34:52.3844 tcp(6) S 192.168.1.1 51336 192.168.1.8 80
2013-12-22-16:34:53.5793 tcp(6) S 192.168.1.1 51334 192.168.1.8 80
2013-12-22-16:34:53.5913 tcp(6) S 192.168.1.1 51335 192.168.1.8 80
2013-12-22-16:34:54.8786 tcp(6) S 192.168.1.1 51337 192.168.1.8 80
2013-12-22-16:34:55.8139 tcp(6) S 192.168.1.1 51338 192.168.1.8 80
2013-12-22-16:34:56.9848 tcp(6) S 192.168.1.1 51339 192.168.1.8 80
2013-12-22-16:34:58.2970 tcp(6) S 192.168.1.1 51340 192.168.1.8 80
2013-12-22-16:35:18.7355 tcp(6) E 192.168.1.1 51345 192.168.1.8 80: 0 512
2013-12-22-16:35:18.3260 honeyd log stopped ------
```

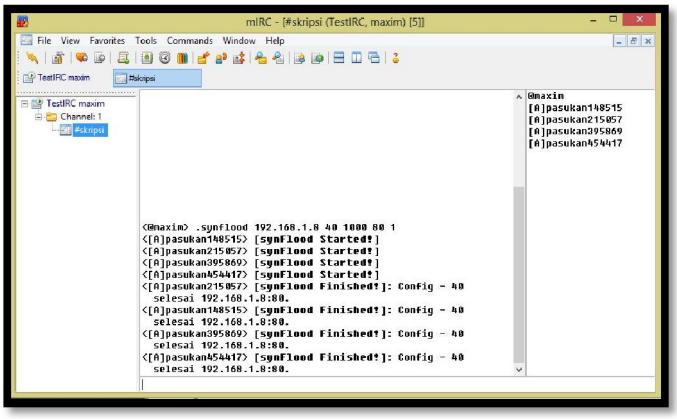
Gambar 4.53 Tampilan file log web emulasi Honeyd terhadap akses normal

Berdasarkan gambar 4.53 merupakan tampilan file log Honeyd terhadap akses normal. Terlihat hanya Alamat IP 192.168.1.1 yang melakukan akses ke web emulasi Honeyd, juga terdapat perbedaan waktu 1 sampai 2 detik saat terjadinya event tersebut. Hal ini membuktikan bahwa user tersebut hanya melakukan akses normal, karena ada perbedaan waktu tersebut. Untuk memahami file log seperti gambar 4.53 cukup sulit, Oleh karena itu dari file log tersebut akan diolah menjadi tampilan grafis seperti gambar 4.54.

	Source IP	↓ Packets	Bytes received
1	192.168.1.1	8	512 B
	Total	8	512 B

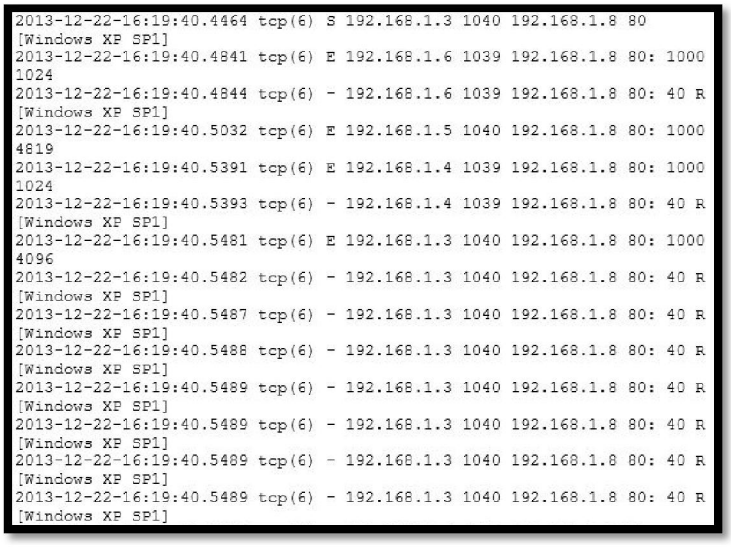
Gambar 4.54 Tampilan grafis setelah user melakukan akses normal

Berdasarkan gambar 4.54 pada tampilan grafis tersebut dapat menjelaskan bahwa saat user melakukan akses normal jumlah paket yang terkirim sebanyak 8 paket dan besar paket yang diterima oleh Honeyd sebesar 512 Byte. Setelah pendeteksian, dan mengolah file log pada Honeyd maka sekarang tibalah saatnya untuk melancarkan serangan DDoS SYN Flood.



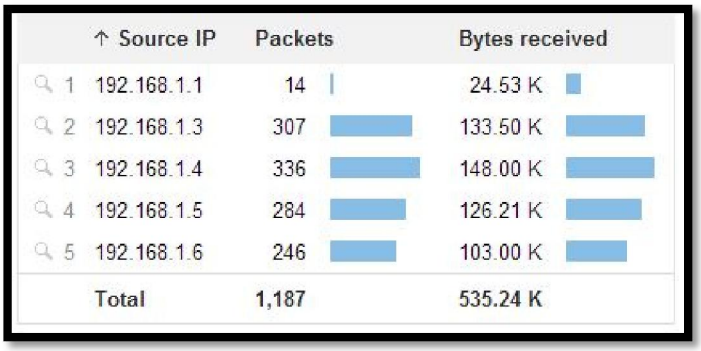
Gambar 4.55 Menjalankan serangan SYN Flood untuk analisa

Pada gambar 4.55 merupakan tampilan botnet pada Mirc, ke-empat botnet tersebut akan diperintah untuk menyerang IP 192.168.1.8 pada Honeyd. Penyerangan tersebut mengirim 40 paket, besar paket 1000 Byte pada port 80 dan waktu tunggu 1 detik. Setelah menjalankan serangan SYN Flood untuk analisa maka selanjutnya pada file log Honeyd akan mencatat serangan-serangan tersebut, berikut adalah file log dari Honeyd.



Gambar 4.56 Tampilan log Honeyd terhadap serangan SYN Flood

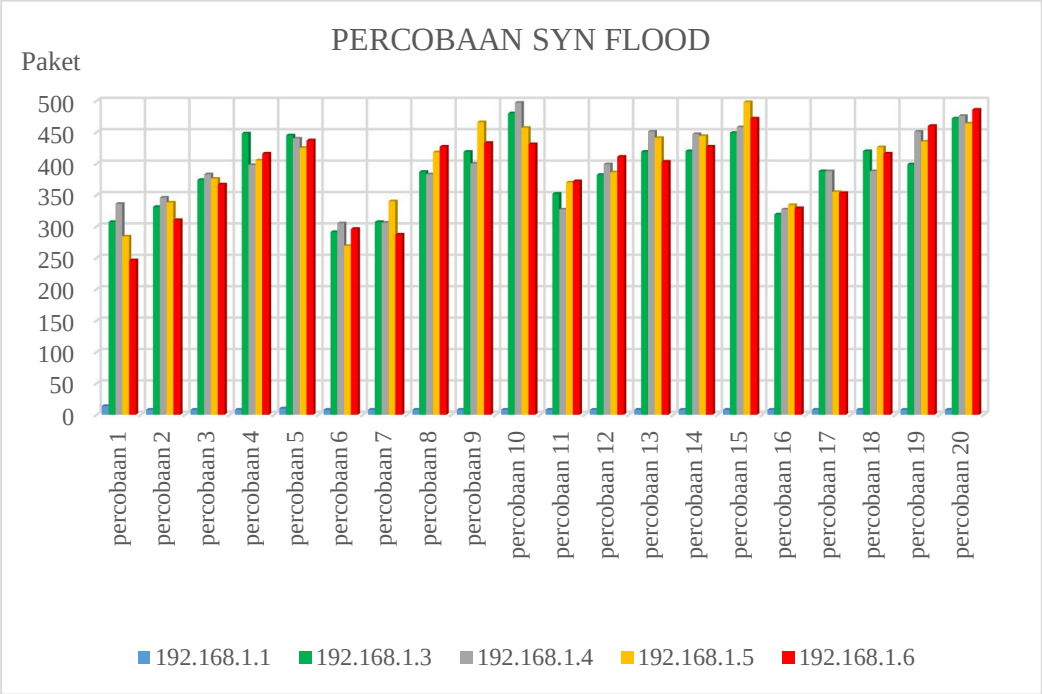
Pada gambar 4.56 merupakan potongan log pada Honeyd ketika serangan SYN Flood dijalankan. untuk memahami log tersebut akan diolah menjadi tampilan grafis seperti pada gambar 4.57.



	↑ Source IP	Packets	Bytes received
1	192.168.1.1	14	24.53 K
2	192.168.1.3	307	133.50 K
3	192.168.1.4	336	148.00 K
4	192.168.1.5	284	126.21 K
5	192.168.1.6	246	103.00 K
	Total	1,187	535.24 K

Gambar 4.57 Tampilan grafis sesudah serangan DDoS SYN Flood

Pada gambar 4.57 terjadi perbedaan yang sangat jauh antara perbedaan jumlah paket yang terkirim pada setiap alamat IP yang mengakses Honeyd dan besar paket yang diterima oleh Honeyd. untuk lebih membuktikan hasil analisa serangan DDoS SYN Flood akan dilakukan percobaaan sebanyak 20 kali percobaan. Berikut pada gambar 4.58 merupakan tampilan diagram untuk setiap 20 kali percobaan serangan DDoS SYN Flood.



Gambar 4.58 Jumlah paket serangan DDoS SYN Flood percobaan 1 sampai 20

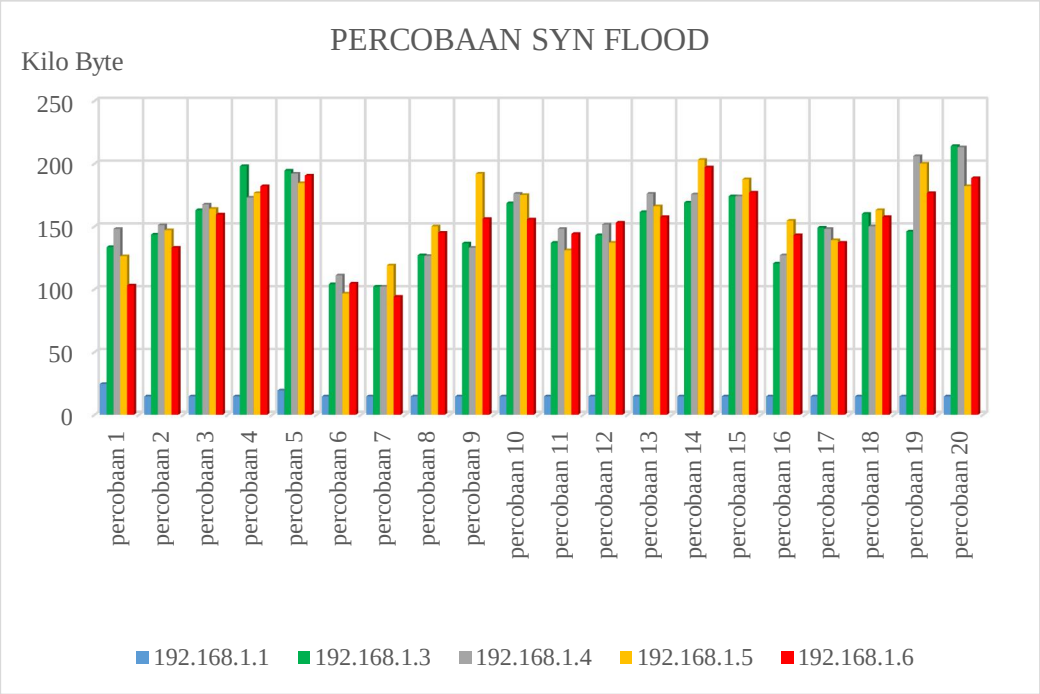
Untuk memudahkan dalam membaca jumlah paket yang diterima oleh Honeyd maka akan ditabelkan seperti tabel 4.3.

Tabel 4.3 Jumlah paket serangan DDoS SYN Flood percobaan 1 sampai 20

PERCOBAAN SYN FLOOD	ALAMAT IP				
	192.168.1.1	192.168.1.3	192.168.1.4	192.168.1.5	192.168.1.6
1	14	307	336	284	246
2	8	331	346	338	310
3	8	374	383	376	367
4	8	448	398	405	416
5	10	445	440	425	437
6	8	291	305	269	296
7	8	307	306	340	287
8	8	387	383	418	427
9	8	419	400	466	433
10	8	480	497	457	431
11	8	352	327	370	372
12	8	382	399	386	411
13	8	419	451	441	403
14	8	420	447	444	427
15	8	449	458	498	472
16	8	319	327	334	329
17	8	388	388	355	353
18	8	420	388	426	416
19	8	399	451	435	460

20	8	472	476	464	486

Dari gambar dan tabel tersebut saat User melakukan akses normal dengan alamat IP 192.168.1.1 pada Honeyd menerima paket yang sedikit yaitu satuan sampai puluhan dan saat Attacker dengan Alamat IP 192.168.1.3 sampai 192.168.1.6 pada Honeyd menerima ratusan paket. Setelah dilakukan 20 kali percobaan serangan DDoS SYN Flood pada web emulasi Honeyd, sekarang saatnya untuk mengetahui besar paket yang diterima oleh Honeyd, berikut adalah tampilan diagram untuk setiap 20 kali percobaan berdasarkan besar paket serangannya.



Gambar 4.59 Besar paket serangan DDoS SYN Flood percobaan 1 sampai 20

Untuk memudahkan dalam membaca besar paket yang diterima oleh Honeyd, maka akan ditabelkan seperti pada tabel 4.4.

Tabel 4.4 Besar paket serangan DDoS SYN Flood percobaan 1 sampai 20

PERCOBAAN	ALAMAT IP				

SYN FLOOD	192.168.1.1	192.168.1.3	192.168.1.4	192.168.1.5	192.168.1.6
1	24,53 KB	133,5 KB	148 KB	126,2 KB	103 KB
2	14,62 KB	143,5 KB	151 KB	147 KB	133 KB
3	14,62 KB	163 KB	167,5 KB	164 KB	159,5 KB
4	14,62 KB	198 KB	173 KB	176,5 KB	182 KB
5	19,32 KB	194,5 KB	192 KB	184,5 KB	190,5 KB
6	14,62 KB	104 KB	111 KB	96,5 KB	104,5 KB
7	14,62 KB	102 KB	102 KB	119 KB	94 KB
8	14,62 KB	127 KB	126,5 KB	150 KB	145 KB
9	14,62 KB	136,5 KB	133 KB	192 KB	156 KB
10	14,62 KB	168,5 KB	176 KB	175 KB	155,5 KB
11	14,62 KB	137 KB	148 KB	131 KB	144 KB
12	14,62 KB	143 KB	151,5 KB	137 KB	153 KB
13	14,62 KB	161,5 KB	176 KB	166 KB	157,5 KB
14	14,62 KB	169 KB	175,5 KB	203 KB	197 KB
15	14,62 KB	174 KB	174 KB	187,5 KB	177 KB
16	14,62 KB	120,5 KB	127 KB	154,5 KB	143 KB
17	14,62 KB	149 KB	148 KB	139 KB	137 KB
18	14,62 KB	160 KB	150 KB	163 KB	157,5 KB
19	14,62 KB	146 KB	206 KB	200 KB	176,5 KB
20	14,62 KB	214 KB	213 KB	182 KB	188,5 KB

Dari gambar dan tabel tersebut saat User melakukan akses normal dengan alamat IP 192.168.1.1, Honeyd menerima besar paket cukup sedikit dibandingkan saat Attacker dengan Alamat IP 192.168.1.3 sampai 192.168.1.6 yang mengirimkan besar paket yang sangat besar ke Honeyd. Setelah semua data telah dikumpulkan

saatnya untuk menyamakan dengan parameter keberhasilan yang sudah ditetapkan.

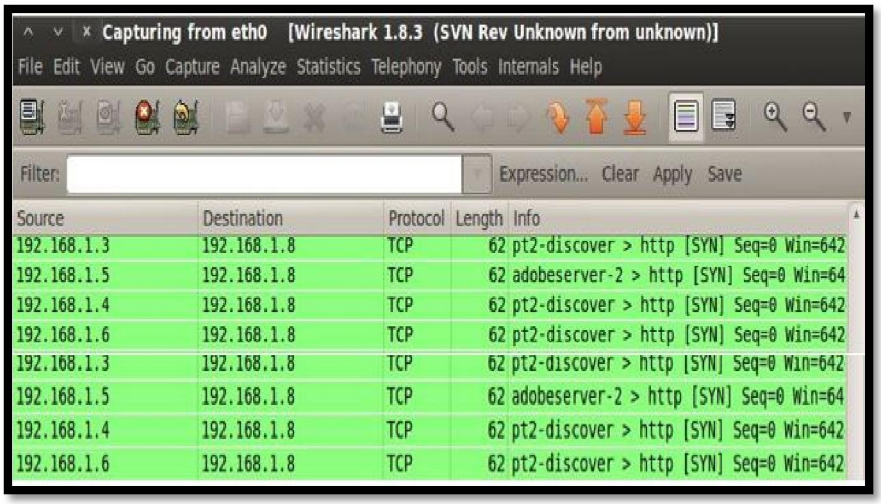
1. Jumlah paket yang diterima oleh Honeyd saat User melakukan akses normal sedikit, sedangkan jumlah paket yang diterima oleh Honeyd saat Attacker melakukan serangan DDoS SYN Flood sangat banyak, hal ini bisa dibuktikan dengan tabel yang sudah dibuat untuk 20 kali serangan SYN Flood.
2. Besar paket yang diterima oleh Honeyd saat User melakukan akses normal sedikit sedangkan besar paket yang diterima oleh Honeyd saat Attacker melakukan serangan DDoS SYN Flood sangat besar.
3. Pada file log Honeyd terhadap serangan DDoS SYN Flood pada gambar 4.56, terdapat banyak alamat IP mulai dari 192.168.1.3, 192.168.1.4, 192.168.1.5 dan 192.168.1.6 yang melakukan akses ke web emulasi Honeyd pada waktu yang relatif sama yaitu pada pukul 16.19.40, hampir dipastikan tidak ada perbedaan waktu sedikitpun saat ke-empat alamat IP tersebut melakukan akses ke web emulasi Honeyd. Dengan demikian bisa dipastikan bahwa ke-empat alamat IP tersebut digunakan oleh Attacker untuk menyerang web emulasi Honeyd.

Untuk membuktikan persentase dari jumlah paket dan besar paket yang diterima oleh Honeyd saat akses normal dan serangan DDoS SYN Flood, maka akan dibuat persentase berdasarkan percobaan selama 20 kali :

a. Total persentase jumlah paket dari akses normal = 0,53%
b. Total persentase jumlah paket dari serangan DDoS SYN Flood = 99,45%
c. Total persentase besar paket dari akses normal =2,40%

d. Total persentase besar paket dari serangan DDoS SYN Flood = 97,58%

Dari ke-tiga ulasan dan persentase tersebut sesuai dengan parameter keberhasilan yang sudah ditetapkan, Hal ini membuktikan bahwa serangan DDoS SYN Flood pada web emulasi honeyd terbukti dan benar adanya. Untuk mengetahui saat proses Penyerangan DDoS SYN Flood yang lebih spesifik ke Honeyd maka digunakan aplikasi wireshark.



Gambar 4.60 Tampilan wireshark ketika dijalankan serangan DDoS SYN Flood

Berdasarkan gambar 4.60 merupakan tampilan wireshark ketika dijalankan serangan DDoS SYN Flood, dimana terdapat beberapa paket SYN yang dikirim oleh alamat IP 192.168.1.3, 192.168.1.5, 192.168.1.4, 192.168.1.6 ke web emulasi Honeyd dengan alamat IP 192.168.1.8.

BAB V

KESIMPULAN DAN SARAN

5.1 KESIMPULAN

Pada sub bab ini akan dibahas tentang kesimpulan seputar penelitian skripsi berdasarkan pada perumusan masalah, manfaat dan tujuan pada bab 1, berikut adalah kesimpulan pada percobaan yang telah dilakukan.

1. Honeyd dapat mengemulasikan web server, virtual host dan beberapa servis yang mirip dengan komputer asli.
2. Honeyd dapat mendeteksi serangan DDOS HTTP Flood dan SYN Flood secara real time. Tetapi untuk lebih membuktikan pola jenis serangan tersebut digunakan aplikasi Wireshark.
3. Honeyd dapat membuat file log yang berisi tentang semua interaksi yang telah ditujukan padanya. Pada file log tersebut terdapat informasi yang cukup lengkap yaitu meliputi alamat IP asal, alamat IP tujuan, port asal, port tujuan, jenis protokol yang digunakan, jumlah paket dan besar paket yang diterima oleh Honeyd, hal ini dapat digunakan untuk membandingkan traffic normal dan traffic serangan.
4. Pada penelitian skripsi ini Honeyd bisa mendeteksi jumlah paket dan besar paket yang diterima oleh Honeyd berdasarkan 20 kali percobaan pada masing-masing jenis serangan, dengan persentase akses normal 2,78% sedangkan serangan DDoS HTTP Flood dan SYN Flood dengan persentase 97,21%

5.2 SARAN

Honeypot merupakan aplikasi yang bisa terus berkembang demi terciptanya sebuah sistem yang lebih baik lagi, dengan perkembangan-perkembangan yang dilakukan bukan tidak mungkin akan menjadikan Honeypot sebagai suatu aplikasi yang powerfull, mengemulasikan web server dengan lebih baik, dan memudahkan bagi seseorang yang ingin mempelajarinya. Adapun saran dari penulis yang ingin melakukan penelitian tentang Honeypot adalah sebagai berikut:

1. Jika tujuan penelitian hanya sekedar mendapatkan informasi-informasi tentang serangan DDoS berdasarkan pada jumlah paket dan besar paket maka bisa menggunakan Honeyd.
2. Untuk mendapatkan sistem keamanan yang lebih baik gunakanlah High Interaction Honeypot.
3. Banyak aplikasi Honeypot dengan jenis yang berbeda dan dengan kemampuan yang berbeda pula sehingga dalam penelitian dapat memilih aplikasi yang diinginkan.

DAFTAR PUSTAKA

Brenton , C., Hunt C. 2005. “Network Security”. PT Elex Media Komputindo. Jakarta.

Holmes, David. 2012. “White Paper of The DDoS Threat Spectrum”. F5 Network, inc.

NSFOCUS. “Common DDoS Attack”. Trademark of NSFOCUS information technology Co., Ltd.

Referens 1, Anonim, Network Security (Keamanan Jaringan), (http://id.wikipedia.org/wiki/Keamanan_jaringan). Diakses pada November 2013

Referens 2, Anonim, DDoS (Distributed Denial of Service) (http://id.wikipedia.org/wiki/Serangan_DoS). Diakses pada November 2013.

Referens 3, Anonim, TCP (Transmission Control Protocol) Flood (http://id.wikipedia.org/wiki/SYN_flooding_attack). Diakses pada November 2013.

Syafrizal, Melwin. 2005. Pengantar Jaringan Komputer. Penerbit Andi. Yogyakarta

Stephen M. Specht and Ruby B. Lee. 2004. “Distributed Denial of Service: Taxonomies of Attacks, Tools, and Countermeasures”. International Workshop on Security in Parallel and Distributed Systems.

Utdirartatmo, Firrar. 2005. “Trik Menjebak Hacker Dengan Honeypot”. Andi Publisher. Yogyakarta.